IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES
(Attorney Docket № 15474US01)

In the Application of:	***Electronically filed on 4-NOV-2009***
Manoj Kumar Singhal	
Serial № 10/803,420	
Filed: March 18, 2004	
For: SYSTEM AND METHOD FOR TIME DOMAIN AUDIO SPEED UP, WHILE MAINTAINING PITCH	
Examiner: COLUCCI, MICHAEL C	
Group Art Unit: 2626	
Confirmation № 5543	
Customer № 23446	

APPEAL BRIEF

Mail Stop Appeal Brief – Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This is an appeal from an Office Action dated June 4, 2009 ("Final Office Action"), in which claims 1-18 were finally rejected. The Appellant respectfully requests that the Board of Patent Appeals and Interferences ("Board") reverses the final rejection of claims 1-18 of the present application. The Appellant notes that this Appeal Brief is timely filed within the two-month period for reply that ends on **November 4, 2009** (the Office date of receipt of the Notice of Appeal being September 4, 2009).

REAL PARTY IN INTEREST
(37 C.F.R. § 41.37(c)(1)(i))

Broadcom Corporation, a corporation organized under the laws of the state of California, and having a place of business at 5300 California Avenue, Irvine, California 92617, has acquired the entire right, title and interest in and to the invention, the application, and any and all patents to be obtained therefor, as set forth in the Assignments recorded at Reel 015075, Frame 0264 in the PTO Assignment Search room.

RELATED APPEALS AND INTERFERENCES
(37 C.F.R. § 41.37(c)(1)(ii))

The Appellant is unaware of any related appeals or interferences.

STATUS OF THE CLAIMS
(37 C.F.R. § 41.37(c)(1)(iii))

Claims 1-18 were finally rejected in the Final Office Action mailed June 4, 2009. Pending claims 1-18 are the subject of this appeal.

The present application includes claims 1-18, which are pending in the present application. Claims 1, 4-6, 9-11 and 14-18 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 5,781,696, by Oh et al. ("Oh"), in view of U.S. Patent No. 6,915,263, by Chen et al. ("Chen"). *See* Final Office Action at pages 4-9.

Claims 2-3, 7-8 and 12-13 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 5,781,696, by Oh et al. ("Oh"), in view of U.S. Patent

No. 6,915,263, by Chen et al. ("Chen"), and further in view of U.S. Patent No. 5,684,829, by Kizuki et al. ("Kizuki"). *See* Final Office Action at pages 9-11.

The Appellant identifies claims 1-18 as the claims that are being appealed. The text of the pending claims is provided in the Claims Appendix.

STATUS OF AMENDMENTS
(37 C.F.R. § 41.37(c)(1)(iv))

The Appellant has not amended any claims subsequent to the final rejection of claims 1-18 mailed on June 4, 2009.

SUMMARY OF CLAIMED SUBJECT MATTER
(37 C.F.R. § 41.37(c)(1)(v))

Independent claim 1 recites the following:

A method for speeding up an encoded original audio signal, said original audio signal having an original frequency and original playback speed,[1] said method comprising:

receiving the encoded original audio signal;[2]

retrieving frames of the original audio signal;[3]

[1] *See* present application, *e.g.*, at page 3, lines 2-5; Figure 2, Figure 3; Figure 5; Figure 6; Figure 7.

[2] *See* present application, *e.g.*, at page 3, lines 26-27; page 7, line 32 – page 8, line 3; page 10, lines 16-20; page 12, lines 11-13; page 13, lines 11-12; Figure 2 (213); Figure 3 (421); Figure 5 (209); Figure 6 (401); Figure 7 (301).

[3] *See id.*, *e.g.*, at page 3, lines 27-28; page 8, lines 2-3; page 11, lines 1-4; page 12, lines 20-23; page 14, lines 7-10; Figure 5 (203); Figure 6 (407); Figure 7 (323).

skipping frames at a rate according to a desired playback speed;[4]

wherein said desired playback speed is greater than the original playback speed;[5]

applying a window function to the remaining frames;[6]

converting the signal with the windowed frames from digital to analog format;[7] and

using the original frequency to playback the analog format signal.[8]

Claims 2-5 and 16 are dependent upon claim 1.

Independent claim 6 recites the following:

A machine-readable storage having stored thereon, a computer program having at least one code section that speed up an encoded original audio signal, said original audio signal having an original frequency and original playback speed, the at least one

---

[4] *See id., e.g.,* at page 3; lines 28-29; page 8, lines 3-7; page 11, lines 8-19; page 12, lines 23-27; page 14, lines 14-28; Figure 2 (212); Figure 3 (423); Figure 5 (202); Figure 6 (409); Figure 7 (325).

[5] *See id., e.g.,* at page 3, lines 12-14; page 8, lines 3-7; page 11, lines 8-19; page 12, lines 23-27; page 14, lines 14-28; Figure 2 (212); Figure 3 (423); Figure 5 (202); Figure 6 (409).

[6] *See id., e.g.,* at page 3, lines 29-30; page 8, lines 8-15; page 11, lines 20-27; page 12, line 28 – page 13, line 3; page 14, line 29 – page 15, line 5; Figure 2 (214); Figure 3 (425); Figure 5 (204); Figure 6 (410); Figure 7 (325).

[7] *See id., e.g.,* at page 3, line 30 – page 4, line 1; page 8, lines 16-20; page 11, lines 28-30; page 13, lines 4-8; page 15, lines 6-9; Figure 3 (427); Figure 5 (201); Figure 6 (411); Figure 7 (327).

[8] *See id., e.g.,* at page 4, lines 1-2; page 8, lines 16-20; page 12, lines 1-7; page 13, lines 4-8; page 15, lines 10-14; Figure 2 (211); Figure 5 (201).

code section being executable by a machine for causing the machine to perform operations[9] comprising:

receiving the encoded original audio signal;[10]

retrieving frames of the original audio signal;[11]

skipping frames at a rate according to a desired playback speed;[12]

wherein said desired playback speed is greater than the original playback speed;[13]

applying a window function to the remaining frames;[14]

converting the signal with the windowed frames from digital to analog format;[15] and

using the original frequency to playback the analog format signal.[16]

---

[9] *See* present application, *e.g.*, at page 3, lines 2-9; Figure 2, Figure 3; Figure 5; Figure 6; Figure 7.

[10] *See id.*, *e.g.*, at page 3, line 10; page 7, line 32 – page 8, line 3; page 10, lines 16-20; page 12, lines 11-13; page 13, lines 11-12; Figure 2 (213); Figure 3 (421); Figure 5 (209); Figure 6 (401); Figure 7 (301).

[11] *See id.*, *e.g.*, at page 3, lines 10-11; page 8, lines 2-3; page 11, lines 1-4; page 12, lines 20-23; page 14, lines 7-10; Figure 5 (203); Figure 6 (407); Figure 7 (323).

[12] *See id.*, *e.g.*, at page 3; lines 11-12; page 8, lines 3-7; page 11, lines 8-19; page 12, lines 23-27; page 14, lines 14-28; Figure 2 (212); Figure 3 (423); Figure 5 (202); Figure 6 (409); Figure 7 (325).

[13] *See id.*, *e.g.*, at page 3, lines 12-14; page 8, lines 3-7; page 11, lines 8-19; page 12, lines 23-27; page 14, lines 14-28; Figure 2 (212); Figure 3 (423); Figure 5 (202); Figure 6 (409).

[14] *See id.*, *e.g.*, at page 3, line 14; page 8, lines 8-15; page 11, lines 20-27; page 12, line 28 – page 13, line 3; page 14, line 29 – page 15, line 5; Figure 2 (214); Figure 3 (425); Figure 5 (204); Figure 6 (410); Figure 7 (325).

[15] *See* present application, *e.g.*, at page 3, lines 15-16; page 8, lines 16-20; page 11, lines 28-30; page 13, lines 4-8; page 15, lines 6-9; Figure 3 (427); Figure 5 (201); Figure 6 (411); Figure 7 (327).

[16] *See id.*, *e.g.*, at page 3, lines 16-17; page 8, lines 16-20; page 12, lines 1-7; page 13, lines 4-8; page 15, lines 10-14; Figure 2 (211); Figure 5 (201).

Claims 7-10 and 17 are dependent upon claim 6.

Independent claim 11 recites the following:

A system that speeds up an encoded original audio signal, said original audio signal having an original frequency and original playback speed,[17] the system comprising:

at least one controller configured to receive the encoded original audio signal;[18]

the at least one controller configured to retrieve frames of the original audio signal;[19]

the at least one controller configured to skip frames at a rate according to a desired playback speed;[20]

wherein said desired playback speed is greater than the original playback speed;[21]

[17] *See* present application, *e.g.,* at page 3, lines 2-5 and 18; Figure 2, Figure 3; Figure 5; Figure 6; Figure 7.
[18] *See id., e.g.,* at page 3, lines 18-19; page 7, line 32 – page 8, line 3; page 10, lines 16-20; page 12, lines 11-13; page 13, lines 11-12; Figure 2 (213); Figure 3 (421); Figure 5 (209); Figure 6 (401); Figure 7 (301).
[19] *See id., e.g.,* at page 3, lines 18-20; page 8, lines 2-3; page 11, lines 1-4; page 12, lines 20-23; page 14, lines 7-10; Figure 5 (203); Figure 6 (407); Figure 7 (323).
[20] *See* present application, *e.g.,* at page 3; lines 18-21; page 8, lines 3-7; page 11, lines 8-19; page 12, lines 23-27; page 14, lines 14-28; Figure 2 (212); Figure 3 (423); Figure 5 (202); Figure 6 (409); Figure 7 (325).

the at least one controller configured to apply a window function to the remaining frames;[22]

the at least one controller configured to convert the signal with the windowed frames from digital to analog format;[23] and

the at least one controller configured to use the original frequency to playback the analog format signal.[24]

Claims 12-15 and 18 are dependent upon claim 11.

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL (37 C.F.R. § 41.37(c)(1)(vi))

Claims 1, 4-6, 9-11 and 14-18 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 5,781,696, by Oh et al. ("Oh"), in view of U.S. Patent No. 6,915,263, by Chen et al. ("Chen"). *See* Final Office Action at pages 4-9.

Claims 2-3, 7-8 and 12-13 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 5,781,696, by Oh et al. ("Oh"), in view of U.S. Patent

[21] *See id., e.g.,* at page 3, lines 12-14; page 8, lines 3-7; page 11, lines 8-19; page 12, lines 23-27; page 14, lines 14-28; Figure 2 (212); Figure 3 (423); Figure 5 (202); Figure 6 (409).
[22] *See id., e.g.,* at page 3, lines 18-19 and 22; page 8, lines 8-15; page 11, lines 20-27; page 12, line 28 – page 13, line 3; page 14, line 29 – page 15, line 5; Figure 2 (214); Figure 3 (425); Figure 5 (204); Figure 6 (410); Figure 7 (325).
[23] *See id., e.g.,* at page 3, lines 18-19 and 23-24; page 8, lines 16-20; page 11, lines 28-30; page 13, lines 4-8; page 15, lines 6-9; Figure 3 (427); Figure 5 (201); Figure 6 (411); Figure 7 (327).
[24] *See id., e.g.,* at page 3, lines 18-19 and 24-25; page 8, lines 16-20; page 12, lines 1-7; page 13, lines 4-8; page 15, lines 10-14; Figure 2 (211); Figure 5 (201).

No. 6,915,263, by Chen et al. ("Chen"), and further in view of U.S. Patent No. 5,684,829, by Kizuki et al. ("Kizuki"). *See* Final Office Action at pages 9-11.

ARGUMENT
(37 C.F.R. § 41.37(c)(1)(vii))

In the Final Office Action, claims 1-18 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over various combinations of Oh, Chen and Kizuki.

I. The Proposed Combination of Oh and Chen Does Not Render Claims 1, 4-6, 9-11 and 14-18 Unpatentable

The Appellant turns to the rejection of claims 1, 4-6, 9-11 and 14-18 as being unpatentable over Oh in view of Chen.

A. Rejection of Independent Claims 1, 6 and 11

With regard to the rejection of independent claims 1, 6 and 11 under 103(a), the Appellant submits that the combination of references cited in the Final Office Action fails to disclose, for example, at least the limitations of "skipping frames at a rate according to a desired playback speed...applying a window function to the remaining frames," as recited in Appellant's independent claims 1 and 6; and "the at least one controller configured to skip frames at a rate according to a desired playback speed...the at least one controller configured to apply a window function to the remaining frames," as recited in Appellant's independent claim 11.

With regard to "skip[ping] frames at a rate according to a desired playback speed...apply[ing] a window function to the remaining frames," the Final Office Action acknowledges that Oh fails to teach applying a window function to the remaining

frames. (Final Office Action, Page 5, Lines 1-2). However, the Final Office Action alleges that Chen's disclosure of muting frames by applying an attenuation function or window to the frame to soften the mute discloses the Appellant's claim limitations. (Final Office Action, Page 5, Line 16 – Page 6, Line 13). The Appellant notes that Chen fails to remedy the deficiencies of Oh for several reasons.

First, the Appellant notes that Chen fails to disclose skipping frames. Thus, Chen also fails to disclose "apply[ing] a window function **to the remaining frames**." The Final Office Action alleges that Chen's disclosure of muting frames teaches skipping frames; however, the Appellant notes that muting frames and skipping frames are different. For example, claim 1 recites "[a] method for speeding up an encoded original audio signal...comprising...skipping frames at a rate according to a desired playback speed." In other words, the playback speed is sped up by skipping frames. Put another way, as defined in the Appellant's claims, skipped frames are not played back in order to speed up a playback speed. Chen's muting has no effect on the playback speed. Rather, Chen's muted frames are played back and the number of muted frames merely impacts the length of the silence period. (*See e.g.*, Chen Column 2, Lines 54-64 and Column 7, Lines 15-20).

Although, in one instance, Chen does refer to a muted frame as "skipped," the Appellant notes that nowhere in Chen is there any disclosure regarding not playing back a muted frame. If Chen did not play back the muted frames (and instead those frames are "skipped" as suggested by the Examiner), then there would be no silence period

and Chen would be rendered inoperable. Therefore, the context of the term "skipped" used in the one instance in Chen is clearly different than the terms "skip" and "skipping" as defined in Appellant's independent claims 1, 6 and 11. Thus, because Chen clearly fails to disclose skipped frames (i.e., frames not played back), Chen fails to remedy the deficiencies of Oh in that the combination of references cannot disclose "apply[ing] a window function **to the remaining frames**," as set forth in Appellant's independent claims 1, 6 and 11.

Second, even if Chen's muted frames could be considered skipped frames (which they clearly are not), the Appellant notes that Chen discloses applying the attenuation function or window to the current frame (i.e., error frame or frame to be muted) to soften the mute. For example, Chen discloses receiving a current frame; determining whether a first error sum is greater than zero for the current frame; if the first error sum is not greater than zero, performing a normal decode of the current frame; if the first error sum is greater than zero, determining whether a second error sum is greater than a tolerance value; if the second error sum is less than the tolerance value, performing a normal decode of the current frame; and **if the second error sum is greater than the tolerance level, muting the current frame by, for example, "apply[ing] a soft mute to the current frame"** or applying a frame repeat process. (Chen, Figure 4 and Column 7, Line 9 through Column 10, Line 26 (emphasis added)).

With regard to applying a soft mute to the current frame, Chen discloses "an attenuation function or 'window' was **applied to the error frame** to soften the mute." (Chen, Column 2, Lines 14-15 (emphasis added)). Chen further discloses "a number of different muting operations can be performed **to mute the current frame**. In the preferred embodiment, a smooth muting with zeros can be applied to decline the audio signal at a given rate **according to a window function**...." (Chen, Column 9, Lines 23-27 (emphasis added)). The Appellant further notes that Chen's Figure 6 "illustrates an audio signal in accordance with **a muted audio frame and a smoothing window function applied thereto**." (Chen, Column 4, Lines 50-52 (emphasis added)).

As clearly shown above, Chen discloses applying its muting/attenuation/smoothing window function to frames that are to be muted. The Final Office Action acknowledges that Chen teaches applying a window function to zero or mute frames. (*See e.g.*, Page 2, Line 21 – Page 3, Line 3). Thus, because Chen clearly discloses applying a window function to mute frames and the Final Office Action acknowledges that the window function is Chen is used to mute or zero frames, the Appellant notes that Chen's teaching of muting frames using a window function is different than "**skipping frames** at a rate according to a desired playback speed... applying a window function **to the remaining frames**," as recited in Appellant's independent claims 1 and 6; and "the at least one controller configured to **skip frames** at a rate according to a desired playback speed... the at least one controller configured to apply a window function **to the remaining frames**," as recited in Appellant's

independent claim 11. Specifically, if Chen's muted frames are to be considered skipped frames (which they clearly are not), the Appellant notes that applying a window function to mute or zero frames is different than "**skip[ping] frames** at a rate according to a desired playback speed...apply[ing] a window function **to the remaining frames**," as set forth in Appellant's independent claims 1, 6 and 11.

Third, the Appellant notes that the Final Office Action states that "applying a window function as taught by Chen to allow for the smoothing of a signal after certain frames were removed/muted." (Final Office Action, Page 6, Lines 16-17). However, the Appellant notes that nowhere in Chen is there any disclosure regarding removing frames. Rather, as noted above, Chen's muted frames are played back as silence periods. Further, the Appellant notes that Chen does not teach applying a window function **after** frames are muted. Rather, Chen teaches applying a window function to a frame to mute the frame. Thus, the Appellant notes that the Final Office Action mischaracterizes Chen's teachings.

Fourth, the Appellant notes that one of ordinary skill in the art would not combine the teachings of Oh and Chen because Oh is related to speed-variable audio playback by adding or deleting separated speech source components of an input audio signal while Chen provides an audio decoder unit that mutes error frames and merges nearby muted frames to extend a silence period between the error frames when the error rate is

high. Put another way, Chen is unrelated to speeding up an audio playback speed and Oh is unrelated to muting error frames. Thus, it is unclear how Chen's teaching of muting error frames adds to Oh's disclosure.

The Final Office Action cites to Chen's Column 9, lines 9-38 and Column 10, lines 1-26 as the motivation to combine the references. However, with regard to Chen's Column 10, lines 1-26, the Appellant notes that the cited section of Chen is unrelated to Chen's muting/attenuation/smoothing window function embodiment. Specifically, Chen teaches "[i]n the preferred embodiment, a smooth muting with zeros can be applied to decline the audio signal at a give rate according to a window function and **in an alternate embodiment, a frame repeat can be performed**." Chen's Column 9, line 64 – Column 10, line 26 relates to Chen's frame repeat embodiment and makes no mention of using a window function. Thus, because Chen's Column 10, lines 1-26 as cited by the Final Office Action is unrelated to Chen's muting/attenuation/smoothing window function embodiment, it is unclear how the cited section provides a motivation to use Chen's window function in Oh.

With regard to Chen's Column 9, lines 9-38, the Appellant notes that the cited section merely teaches applying a window function to mute a current frame based on an error rate. As noted above, Oh is unrelated to muting frames based on an error rate. Thus, it is unclear how the cited section provides a motivation to use Chen's window function in Oh.

Basically, the combination of Oh and Chen fail to disclose, for example, at least the limitations of "**skipping frames** at a rate according to a desired playback speed... applying a window function **to the remaining frames**," as recited in Appellant's independent claims 1 and 6; and "the at least one controller configured to **skip frames** at a rate according to a desired playback speed... the at least one controller configured to apply a window function **to the remaining frames**," as recited in Appellant's independent claim 11. Rather, Oh merely discloses applying a window function to the audio characteristics component. Nowhere in Oh is there any disclosure regarding applying a window function to the speech source components not deleted by the speech source modulating unit of the pitch modulating unit 4. Thus, as acknowledged by the Final Office Action, Oh fails to disclose "apply[ing] a window function **to the remaining frames**," as set forth in Appellant's independent claims 1, 6 and 11. Chen fails to remedy the deficiencies of Oh in that Chen merely discloses applying a window function to mute a current frame based on an error rate, which is different than "**skip[ping] frames** at a rate according to a desired playback speed...apply[ing] a window function **to the remaining frames**," as set forth in Appellant's independent claims 1, 6 and 11.

Therefore, the Appellant maintains that at least the limitations "**skipping frames** at a rate according to a desired playback speed... applying a window function **to the remaining frames**," as recited in Appellant's independent claims 1 and 6; and "the at least one controller configured to **skip frames** at a rate according to a desired playback

speed... the at least one controller configured to apply a window function **to the remaining frames**," as recited in Appellant's independent claim 11, are not obvious over Oh in view of Chen. Accordingly, independent claims 1, 6 and 11 are not unpatentable over Oh in view of Chen and are allowable.

B. Examiner's Response to Arguments

The Examiner responded to the Appellant's arguments on pages 2-3 of the Final Office Action.

Specifically, the Final Office Action states the following:

> Chen describes that which is well known in the art. Consider the inherency of a window function is directed to preserving signal data within a window/interval, wherein any data outside the interval is zeroed (or muted for an audio signal, and thus skipped). The concept of Oh is realized through the teaching of Chen, wherein by applying a window function to frames, the zeroing or muting frames is present. Also consider that a window function itself can be applied to a speech signal with given parameters that allow for elimination of data outside a windowed frame (i.e. other frames NOT in the window). Chen thus teaches attenuation of the signal outside the windowed area (Chen Col. 9 lines 9-38). Further, consider the purpose of a window function in a speech signal in the instance where increasing the playback speed alone may have residual undesirable effects. Thus the use of a window function as taught by Chen, would alleviate any residual effects or noise present after skipping frames. This is also consistent with the present invention, wherein both Chen and the present invention teach the concept of overlapping as well as "smoothing" a signal out (present invention [0028]). Chen in explicitly teaches the elimination of errors (residual effects or noise) by the well known use of a window function to completely stop any surrounding errors. Chen also differentiates between the use of partial and full attenuation through window functions (Chen Col. 10 lines 1-26).

(Final Office Action, Page 2, Line – Page 3, Line 17). The Appellant notes, however, that the Final Office Action mischaracterizes Chen and the Appellant's claims.

First, the Appellant notes that Appellant's claims clearly recite that frames are skipped prior to applying a window function (i.e. "applying a window function to the remaining frames"). Thus, the Final Office Action's allegation that frames are skipped or muted **by applying a window function** to non-muted frames is different than "skipping frames at a rate according to a desired playback speed... applying a window function **to the remaining frames**," as recited in Appellant's independent claims 1 and 6; and "the at least one controller configured to skip frames at a rate according to a desired playback speed... the at least one controller configured to apply a window function **to the remaining frames**," as recited in Appellant's independent claim 11.

Second, Chen teaches applying a window function to a current frame to mute the current frame. Thus, if the Final Office Action is interpreting Chen's muted frames to be skipped (despite Chen's teaching that all frames are played back), Chen fails to remedy the deficiencies of Oh in that the combination of references clearly fail to disclose "**skip[ping] frames** at a rate according to a desired playback speed...apply[ing] a window function **to the remaining frames**," as set forth in Appellant's independent claims 1, 6 and 11.

Third, with regard to the Final Office Action's allegation that "Chen and the present invention teach the concept of overlapping as well as 'smoothing' a signal out," the Appellant notes that Chen's overlap teachings are not related to the application of

window functions, and instead are directed to repeating a previous frame to conceal an error in the current frame <u>**in lieu of**</u> **soft muting using a window function**. (*See e.g.*, Chen, Column 9, Lines 24-28 and Lines 64-66; Column 10, Lines 11-19 and 23-26). Nowhere in Chen is there any teaching regarding overlap in connection with performing a window function. Thus, the Appellant notes that the Final Office Action mischaracterizes Chen.

Fourth, with regard to the Final Office Action's allegation that "Chen also differentiates between the use of partial and full attenuation through window functions (Chen Col. 10 lines 1-26)," the Appellant notes, as discussed above, that the cited section of Chen is unrelated to window functions. Rather, the cited section of Chen discusses repeating frames in lieu of soft muting using window functions when there are less than three or four consecutive error frames. (*See e.g.*, Chen, Column 9, Lines 64 – Column 10, Line 7; Column 10, Lines 23-26). Thus, the Appellant notes that the Final Office Action mischaracterizes Chen.

Accordingly, independent claims 1, 6 and 11 are not unpatentable over Oh in view of Chen and are allowable. Furthermore, the Appellant reserves the right to argue additional reasons beyond those set forth herein to support the allowability of claims 1, 6 and 11.

C. Rejection of Dependent Claims 4-5, 9-10 and 14-15

Claims 4-5, 9-10 and 14-15 depend on independent claims 1, 6 and 11, respectively. Therefore, the Appellant submits that claims 4-5, 9-10 and 14-15 are allowable over the combination of references cited in the Final Office Action at least for the reasons stated above with regard to claims 1, 6 and 11.

The Appellant also submits that at least the limitation of "wherein the desired playback speed is **a predefined default value**," as recited by the Appellant in claim 4, 9 and 14; and "wherein the desired playback speed is **a programmable value**," as recited by the Appellant in claims 5, 10 and 15, are not obvious over Oh in view of Chen.

The Final Office Action alleges that Oh's Column 6, Lines 34-38 teaches "wherein the desired playback speed is a predefined default value," as recited by the Appellant in claim 4, 9 and 14; and "wherein the desired playback speed is a programmable value," as recited by the Appellant in claims 5, 10 and 15. The cited section of Oh states the following:

> δq: a variable for determining the play-back speed.
>
> The speed-varied speech signal x(n) is sent to the D/A converter 7 via the output buffer 6. In the D/A converter 7, the speech signal x(n) is converted into an analog signal which is, in turn, output as an audio-out signal.

(Oh, Column 6, Lines 34-38). Clearly, nowhere in the cited section of Oh is there any mention of the playback speed being **a predefined default value**. Nor does the cited section of Oh teach that the desired playback speed is **a programmable value**. Rather, the cited section of Oh merely teaches that δq is a variable for determining play-

back speed and x(n) is a speed-varied speech signal. The Appellant notes that Oh's disclosure fails to teach "wherein the desired playback speed is a predefined default value," as recited by the Appellant in claim 4, 9 and 14; and "wherein the desired playback speed is a programmable value," as recited by the Appellant in claims 5, 10 and 15. Further, Chen fails to remedy the deficiencies of Oh. Accordingly, the Appellant submits that claims 4-5, 9-10 and 14-15 are allowable over the combination of references cited in the Final Office Action at least for the above reasons.

The Appellant also reserves the right to argue additional reasons beyond those set forth above to support the allowability of claims 4-5, 9-10 and 14-15.

D. Rejection of Dependent Claims 16-18

Claims 16, 17 and 18 depend on independent claims 1, 6 and 11, respectively. Therefore, the Appellant submits that claims 16, 17 and 18 are allowable over the combination of references cited in the Final Office Action at least for the reasons stated above with regard to claims 1, 6 and 11.

The Appellant also submits that at least the limitation of "wherein skipping frames at a rate according to a desired playback speed further comprises skipping frames at a rate according to a desired playback speed, wherein the frames **correspond to time intervals**," as recited by the Appellant in claims 16, 17 and 18, are not obvious over Oh in view of Chen.

The Final Office Action alleges that Chen's Column 7, Lines 37-55 and Column 9, Lines 9-38 teaches "wherein skipping frames at a rate according to a desired playback speed further comprises skipping frames at a rate according to a desired playback speed, wherein the frames **correspond to time intervals**," as recited by the Appellant in claims 16, 17 and 18. However, nowhere in the cited section of Chen is there any mention of "wherein skipping frames at a rate according to a desired playback speed further comprises skipping frames at a rate according to a desired playback speed, wherein the frames **correspond to time intervals**," as recited by the Appellant in claims 16, 17 and 18. Further, Oh fails to remedy the deficiencies of Chen. Accordingly, the Appellant submits that claims 16, 17 and 18 are allowable over the combination of references cited in the Final Office Action at least for the above reasons.

The Appellant also reserves the right to argue additional reasons beyond those set forth above to support the allowability of claims 16, 17 and 18.

II. The Proposed Combination of Oh, Chen and Kizuki Does Not Render Claims 2-3, 7-8 and 12-13 Unpatentable

The Appellant turns to the rejection of claims 2-3, 7-8 and 12-13 as being unpatentable over Oh in view of Chen, and further in view of Kizuki.

Claims 2-3, 7-8 and 12-13 depend on independent claims 1, 6 and 11, respectively, and Kizuki fails to remedy the previously mentioned deficiencies of Oh in

view of Chen. Therefore, the Appellant submits that claims 2-3, 7-8 and 12-13 are allowable over the combination of references cited in the Final Office Action at least for the reasons stated above with regard to claims 1, 6 and 11.

Accordingly, the Appellant submits that claims 2-3, 7-8 and 12-13 are allowable over the combination of references cited in the Final Office Action at least for the above reasons. The Appellant also reserves the right to argue additional reasons beyond those set forth above to support the allowability of claims 2-3, 7-8 and 12-13.

CONCLUSION

For at least the foregoing reasons, the Appellant submits that claims 1-18 are in condition for allowance. Reversal of the Examiner's rejection and issuance of a patent on the application are therefore requested.

The Commissioner is hereby authorized to charge $540 (to cover the Brief on Appeal Fee) and any additional fees or credit any overpayment to the deposit account of McAndrews, Held & Malloy, Ltd., Account No. 13-0017.

Respectfully submitted,

Date: 4-NOV-2009

By: /Philip Henry Sheridan/
Philip Henry Sheridan
Reg. No. 59,918
Attorney for Appellant

McANDREWS, HELD & MALLOY, LTD.
500 West Madison Street, 34th Floor
Chicago, Illinois 60661
(T) 312 775 8000
(F) 312 775 8100

(PHS)

CLAIMS APPENDIX
(37 C.F.R. § 41.37(c)(1)(viii))

1. A method for speeding up an encoded original audio signal, said original audio signal having an original frequency and original playback speed, said method comprising:

receiving the encoded original audio signal;

retrieving frames of the original audio signal;

skipping frames at a rate according to a desired playback speed;

wherein said desired playback speed is greater than the original playback speed;

applying a window function to the remaining frames;

converting the signal with the windowed frames from digital to analog format; and

using the original frequency to playback the analog format signal.

2. The method according to claim 1 wherein the encoded original audio signal is encoded in the frequency domain using one of a plurality of encoding schemes, the method further comprising frequency-domain decoding of the encoded original audio signal.

3. The method according to claim 2 wherein said decoding comprises: decoding said encoded signal using a decoding scheme corresponding to said one of a plurality of encoding schemes; applying an inverse transform to the encoded audio signal; and applying an inverse window function.

4. The method according to claim 1 wherein the desired playback speed is a predefined default value.

5. The method according to claim 1 wherein the desired playback speed is a programmable value.

6. A machine-readable storage having stored thereon, a computer program having at least one code section that speed up an encoded original audio signal, said original audio signal having an original frequency and original playback speed, the at least one code section being executable by a machine for causing the machine to perform operations comprising:

receiving the encoded original audio signal;

retrieving frames of the original audio signal;

skipping frames at a rate according to a desired playback speed;

wherein said desired playback speed is greater than the original playback speed;

applying a window function to the remaining frames;

converting the signal with the windowed frames from digital to analog format; and

using the original frequency to playback the analog format signal.

7. The machine-readable storage according to claim 6 wherein the encoded original audio signal is encoded in the frequency domain using one of a plurality of encoding schemes, the machine-readable storage further comprising code for frequency-domain decoding of the encoded original audio signal.

8. The machine-readable storage according to claim 7 further comprising: code for decoding said encoded signal using a decoding scheme corresponding to said one of a plurality of encoding schemes; code for applying an inverse transform to the encoded audio signal; and code for applying an inverse window function.

9. The machine-readable storage according to claim 6 wherein the desired playback speed is a predefined default value.

10. The machine-readable storage according to claim 6 wherein the desired playback speed is a programmable value.

11. A system that speeds up an encoded original audio signal, said original audio signal having an original frequency and original playback speed, the system comprising:

at least one controller configured to receive the encoded original audio signal;

the at least one controller configured to retrieve frames of the original audio signal;

the at least one controller configured to skip frames at a rate according to a desired playback speed;

wherein said desired playback speed is greater than the original playback speed;

the at least one controller configured to apply a window function to the remaining frames;

the at least one controller configured to convert the signal with the windowed frames from digital to analog format; and

the at least one controller configured to use the original frequency to playback the analog format signal.

12. The system according to claim 11 wherein the encoded original audio signal is encoded in the frequency domain using one of a plurality of encoding schemes, the system further comprising code for frequency-domain decoding of the encoded original audio signal.

13. The system according to claim 12 further comprising: the at least one controller configured to decode said encoded signal using a decoding scheme corresponding to said one of a plurality of encoding schemes; the at least one controller configured to apply an inverse transform to the encoded audio signal; and the at least one controller configured to apply an inverse window function.

14. The system according to claim 11 wherein the desired playback speed is a predefined default value.

15. The system according to claim 11 wherein the desired playback speed is a programmable value.

16. The method of claim 1, wherein skipping frames at a rate according to a desired playback speed further comprises skipping frames at a rate according to a desired playback speed, wherein the frames correspond to time intervals.

17. The machine-readable storage of claim 6, wherein skipping frames at a rate according to a desired playback speed further comprises skipping frames at a rate according to a desired playback speed, wherein the frames correspond to time intervals.

18. The system of claim 11, wherein skipping frames at a rate according to a desired playback speed further comprises skipping frames at a rate according to a desired playback speed, wherein the frames correspond to time intervals.

EVIDENCE APPENDIX
(37 C.F.R. § 41.37(c)(1)(ix))

(1) United States Patent No. 5,781,696 ("Oh"), entered into record by the Examiner in the May 25, 2007 Office Action.

(2) United States Patent No. 6,915,263 ("Chen"), entered into record by the Examiner in the October 27, 2008 Office Action.

(3) United States Patent No. 5,684,829 ("Kizuki"), entered into record by the Examiner in the October 27, 2008 Office Action.

RELATED PROCEEDINGS APPENDIX
(37 C.F.R. § 41.37(c)(1)(x))

The Appellant is unaware of any related appeals or interferences.

US005781696A

United States Patent [19]

Oh et al.

[11] **Patent Number:** 5,781,696

[45] **Date of Patent:** Jul. 14, 1998

[54] **SPEED-VARIABLE AUDIO PLAY-BACK APPARATUS**

[75] Inventors: **Yung Hwan Oh**; **Yeon Jun Kim**, both of Taejon; **Jum Han Bae**, Suwon, all of Rep. of Korea

[73] Assignee: **Samsung Electronics Co., Ltd.**, Kyungki-Do, Rep. of Korea

[21] Appl. No.: **535,517**

[22] Filed: **Sep. 28, 1995**

[30] **Foreign Application Priority Data**

Sep. 28, 1994 [KR] Rep. of Korea 1994-24514

[51] **Int. Cl.**[6] **G01L 9/00**

[52] **U.S. Cl.** **395/2.79**; 395/2.16; 395/2.15; 395/2.33

[58] **Field of Search** 395/2.91, 2.92, 395/2.19, 2.15, 2.17, 2.79, 2.16, 2.14, 2.42, 2.33; 381/63, 106

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,164,626	8/1979	Fette	395/2.17
4,624,012	11/1986	Lin et al.	381/51
4,734,795	3/1988	Fukami et al.	360/8
4,783,807	11/1988	Marley	395/2.44
5,298,674	3/1994	Yun	84/616
5,365,592	11/1994	Horner et al.	395/2.59
5,375,188	12/1994	Serikawa et al.	395/2.24
5,457,769	10/1995	Valley	395/2.19

FOREIGN PATENT DOCUMENTS

4168499	6/1992	Japan	G10L 3/02

OTHER PUBLICATIONS

JAPIO abstract of JP57154607A; Dialog(R)File 347:JAPIO; (c) JPO & JAPIO

Primary Examiner—Allen R. MacDonald
Assistant Examiner—Vijay B. Chawan
Attorney, Agent, or Firm—Sughrue, Mion, Zinn, Macpeak & Seas, PLLC

[57] **ABSTRACT**

A speed-variable audio play-back apparatus which includes a pitch detecting circuit for separating speech source components and audio characteristics from an input audio signal, a pitch modulating unit for deleting selected ones of the separated speech source components or adding another speech source components to the separated speech source components depending on a play-back speed, thereby adjusting the length of the audio signal to be played back, a speech synthesizing circuit for synthesizing the speech source components and audio characteristics modulated by the pitch modulating unit, thereby outputting a speed-varied audio signal; and a main controller for controlling the above components in accordance with control signals externally applied thereto. With this arrangement, it is possible to play back audio stored in a storage medium at an adjusted speed while preventing degradation in tone color and loss of audio signals from occurring upon varying the play-back speed when the audio is played back by an apparatus such as a tape player, VTR, multimedia equipment, or computer, so that the played-back audio sounds like a person speaking quickly or slowly.

8 Claims, 4 Drawing Sheets

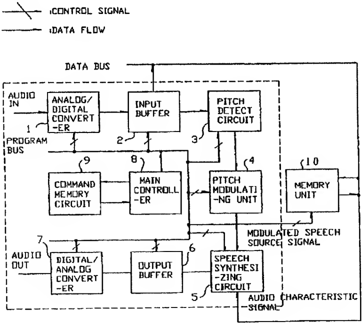


FIG. 1

PRIOR ART

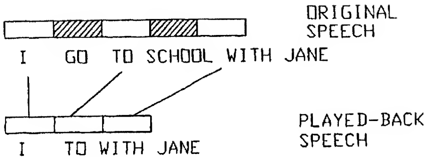


FIG. 3

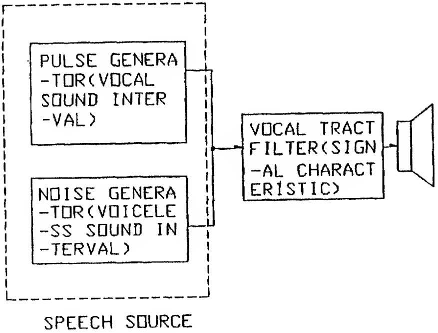


FIG.2

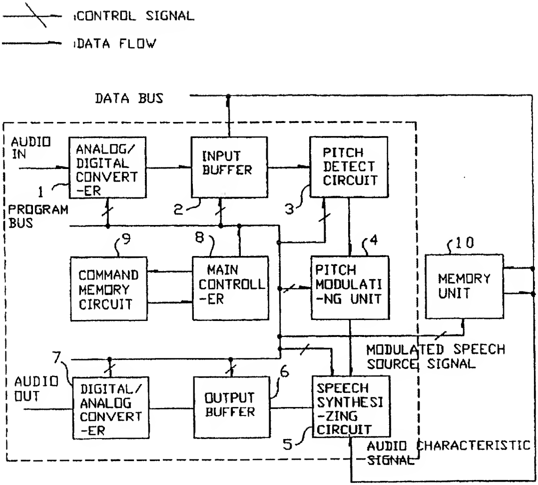

:CONTROL SIGNAL
:DATA FLOW
DATA BUS
AUDIO IN
ANALOG/ DIGITAL CONVERT -ER
1
INPUT BUFFER
2
PITCH DETECT CIRCUIT
3
PROGRAM BUS
9
8
4
10
COMMAND MEMORY CIRCUIT
MAIN CONTROLL -ER
PITCH MODULATI -NG UNIT
MEMORY UNIT
MODULATED SPEECH SOURCE SIGNAL
7
AUDIO OUT
DIGITAL/ ANALOG CONVERT -ER
OUTPUT BUFFER
6
SPEECH SYNTHESI -ZING CIRCUIT
5
AUDIO CHARACTERISTIC SIGNAL

FIG.4

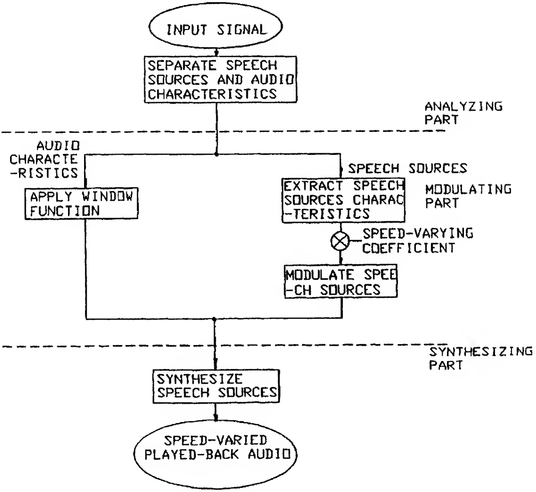

INPUT SIGNAL
SEPARATE SPEECH SOURCES AND AUDIO CHARACTERISTICS
ANALYZING PART
AUDIO CHARACTE-RISTICS
SPEECH SOURCES
APPLY WINDOW FUNCTION
EXTRACT SPEECH SOURCES CHARAC-TERISTICS
MODULATING PART
SPEED-VARYING COEFFICIENT
MODULATE SPEE-CH SOURCES
SYNTHESIZING PART
SYNTHESIZE SPEECH SOURCES
SPEED-VARIED PLAYED-BACK AUDIO

FIG.5

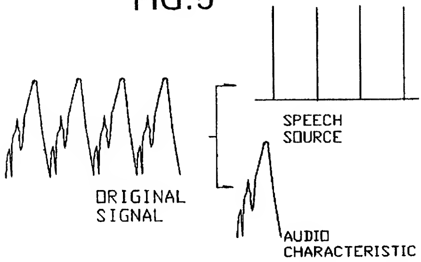
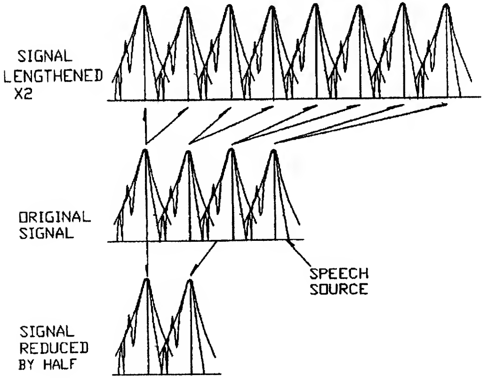

ORIGINAL
SIGNAL
SPEECH
SOURCE
AUDIO
CHARACTERISTIC

FIG.6


SIGNAL
LENGTHENED
X2
ORIGINAL
SIGNAL
SPEECH
SOURCE
SIGNAL
REDUCED
BY HALF

SPEED-VARIABLE AUDIO PLAY-BACK APPARATUS

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to a speed-variable audio play-back apparatus, and more particularly to a speed-variable audio play-back apparatus capable of playing back audio stored in a storage medium at an adjusted speed while preventing any degradation in tone color or loss of audio signals from occurring upon varying the play-back speed while the audio (speech) is played back by an audio play-back apparatus such as a tape player, VTR, multimedia equipment, computer and the like, so that the audio (speech) being played back can be heard as when a person speaks quickly or slowly.

2. Description of Related Art

In tape or video players, generally, the tone color of the audio varies when the play-back speed varies. When play-back is carried out at a high speed, the audio being played back is different from its original audio level, and it is heard as a "peep-peep" sound. At a low play-back speed, a sound typically called "loosened tape sound", is generated.

As a conventional method for preventing such phenomena, Japanese Patent Laid-open Publication No. Heisei 4-168499 (Jun. 16, 1992) discloses a method for partially playing back audio (speech) signals read by a memory buffer. In accordance with this method, when the play-back speed is doubled, audio (speech) signals read by the memory buffer are partially played back such that only one of its two successive time-slices is played back.

For example, if the phrase, "I go to school with Jane", is played back at a double speed using the above-mentioned conventional method, components of the original audio respectively corresponding to the shaded portions shown in FIG. 1 are eliminated, so that only the speech "I to with Jane" is played back.

Since the conventional method plays back only part of the speech at a higher play-back speed so as to keep the tone color of the speech, the original meaning of the speech is lost. As a result, it is very difficult to understand the meaning of the speech using the conventional play-back apparatus. Furthermore, it makes listeners feel uncomfortable.

SUMMARY OF THE INVENTION

Therefore, an object of the invention is to solve the above-mentioned problem and to provide a speed-variable audio play-back apparatus capable of playing back audio stored in a storage medium at an adjusted speed while preventing any degradation in tone color and loss of audio signals from occurring upon varying the play-back speed while the audio (speech) is played back by an audio play-back apparatus such as a tape player, VTR, multimedia equipment, computer and the like, so that the audio (speech) being played back can be heard as when a person speaks quickly or slowly.

In accordance with the present invention, this object is accomplished by providing a speed-variable audio play-back apparatus comprising: a pitch detecting circuit for separating speech source components and audio characteristics from an input audio signal; a pitch modulating unit for deleting selected ones of the separated speech source components or adding another speech source component to the separated speech source components depending on a play-back speed, thereby adjusting the length of the audio signal to be played back; a speech synthesizing circuit for synthesizing the speech source components and audio characteristics modulated by the pitch modulating unit, thereby outputting a speed-varied audio signal; and a main controller for controlling the circuits and unit in accordance with control signals externally applied thereto, respectively.

It is preferred that the pitch detecting circuit be provided with an analog/digital converter for converting the analog input audio signal into a digital audio signal so that the pitch detecting circuit detects pitch portions of the audio signal in a digital manner.

It is also preferred that the speech synthesizing circuit be provided with a digital/analog converter for converting the audio signal, conversion-processed in a digital manner, into an analog signal.

Preferably, the apparatus further comprises a memory unit for temporarily storing the initial audio signal and sending the stored audio signal to the speech synthesizing circuit so that the audio signal is compared with the modulated audio signal synthesized by the speech synthesizing circuit.

It is also preferred that the apparatus further comprises a command memory circuit for storing various control signals required for a speed-varied audio play-back, receiving control signals from the main controller and outputting the stored control signals respectively based on the received control signals.

It is also preferred that the pitch detecting circuit extracts the speech source components on the basis of the following equation:

$$c(m, \delta) = \sum_{n=0}^{N-1} |x(n + t(m-1)) - x(n + tm + \delta)|$$

where,

x(n): the original input signal (the amount of speech on a time axis n);

tm: the position of the m-th speech source; and

δ: a tolerance region around tm.

Preferably, the pitch modulating unit performs a signal modulation by applying a window function, which provides a required signal length extending from the position of each speech source, to a portion of the audio signal corresponding to each audio signal characteristic as expressed by the following equation:

$$x_m(n) = h_m(tm - n)x(n)$$

where,

$x_m(n)$: the modulated audio signal;

$h_m(n)$: the window function;

t_m: the position of each speech source; and

x(n): the input audio signal (the amount of speech on a time axis n).

Preferably, the speech source synthesizing circuit derives a speed-varied speech signal by use of the modulated speech source components and audio signal characteristics as expressed by the following equation:

$$x(n) = \frac{\sum_q \alpha q x q(n) h q^2(tq - n)}{\sum_q h q^2(tq - n)}$$

where,

x(n): the speed-varied speech signal;

αq: a variable for adjusting the amount of synthesized speech;

$xq(n)$: the modulated audio characteristics ($xq(n)=x_m(n-\delta q)$);

tq; the position of each modulated speech source; and

δq: a variable for determining the play-back speed;

BRIEF DESCRIPTION OF THE DRAWINGS

Other objects and aspects of the invention will become apparent from the following description of embodiments with reference to the accompanying drawings in which:

FIG. 1 is a diagram for explaining a conventional speed-variable speech play-back system;

FIG. 2 is a block diagram schematically illustrating a speed-variable audio play-back apparatus in accordance with the present invention;

FIG. 3 is a block diagram illustrating a speech production model, applied to the present invention, in the form of an electronic circuit;

FIG. 4 is a flow chart illustrating signal processing procedures respectively executed by main parts of the speed-variable audio play-back apparatus shown in FIG. 2;

FIG. 5 is a waveform diagram showing waveforms of speech sources and audio characteristics separated in an analyzing procedure executed by the speed-variable audio play-back apparatus of FIG. 2; and

FIG. 6 is a waveform diagram showing a procedure for modulating the speech source by the speed-variable audio play-back apparatus of FIG. 2.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

FIG. 2 is a block diagram schematically illustrating a speed-variable audio play-back apparatus in accordance with the present invention.

As shown in FIG. 2, the apparatus includes an analog/digital (A/D) converter 1 connected to an audio-in line and a program bus. An input buffer 2 is connected to the A/D converter 1. The input buffer 2 is also coupled to a data bus as well as the program bus. The apparatus further includes a pitch detecting circuit 3 connected to both the input buffer 2 and the program bus, a pitch modulating unit 4 connected to both the pitch detecting circuit 3 and the program bus, and a speech synthesizing circuit 5 connected to the pitch modulating unit 4. The speech synthesizing circuit 5 is also coupled to both the program bus and the data bus. The apparatus also includes an output buffer 6 connected to both the speech synthesizing circuit 5 and the program bus, a digital/analog (D/A) converter 7 connected to both the output buffer 6 and the program bus, a main controller 8 connected to the program bus, a command memory circuit 9, such as read only memory (ROM), connected to the main controller 8, and a memory unit 10, such as random access memory (RAM), connected to both the program bus and the data bus.

The main controller 8 serves to control the overall system of the speed-variable audio play-back apparatus. Command languages required to control various parts by the main controller 8 are stored in the command memory circuit 9. On the other hand, audio data is stored in the memory unit 10.

Transfer of control signals and transfer of data among the blocks are carried out by the program bus and data bus, respectively. The program bus serves to transfer a command from the main controller 8 to a part to be controlled. The data bus serves to receive audio data from the input buffer 2 and to temporarily store the received audio signal. Upon a speech synthesis, the data bus sends the stored audio data to the speech synthesizing circuit 5 so that the audio data can be re-synthesized with a modulated speech source signal in the speech synthesizing circuit 5.

Operation of the speed-variable audio play-back apparatus having the above-mentioned arrangement according to the present invention will now be described.

The system used in the apparatus according to the present invention is based on a speech production model which simulates a speaker's vocal organ. In accordance with the speech production model, the audio is determined by an audio transfer characteristic obtained by a speech source, which is an audio production source, and an articulation organ such as a tongue, a lip, or teeth.

In accordance with the speech production model, a flow of air emerging from the speaker's lungs generates periodic or noisy air vibrations in "a narrow space" defined in the voice cord or oral cavity by the tongue, lip, or teeth; that is, at the point of articulation. These air vibrations become a speech source. The frequency component of the speech source is selectively resonated by the influence of the audio transfer characteristic determined by the articulation of an organ positioned above the voice cord, namely, the vocal tract, thereby producing speech.

Referring to FIG. 3, such a speech production model is schematically shown in the form of an electronic circuit.

This system shown in FIG. 3 is based on the above-mentioned speech production model. As shown in FIG. 4, the system includes three main parts, namely, an analyzing part for separating speech sources and audio characteristics from an input signal, a modulating part for processing the separated signals at the desired play-back speed, and a synthesizing part for performing a signal re-synthesis using the modulated signals.

The modulating part includes a speech source modulating unit adapted to process the separated speech source signals based on the above-mentioned speech production model, and an audio characteristic control unit adapted to perform a smoothing process using a window function needed for the re-synthesis while maintaining the tone color, namely, the audio characteristic.

The overall operation of this system is constituted by procedures of analyzing an input audio signal to vary the play-back speed while still maintaining the tone color or frequency of the audio signal, separating speech sources and audio characteristics from the audio signal based on the result of the analysis, processing the separated data at a varied play-back speed, and performing a signal re-synthesis using the processed data. These procedures are best shown in FIG. 4.

FIG. 4 shows signal processing procedures respectively carried out by the main parts of the speed-variable audio play-back apparatus shown in FIG. 2.

The analyzing, modulating and synthesizing parts, which are the most important parts of the present invention, correspond to the pitch detecting circuit 3, the pitch modulating unit 4 and the speech synthesizing circuit 5, respectively.

The above procedures will now be described in more detail in conjunction with the apparatus shown in FIG. 2.

Once an analog audio signal is input, it is converted into a digital signal by the A/D converter 1 and then sent to the pitch detecting circuit 3 via the input buffer 2.

In the procedure executed by the analyzing part, the pitch detecting circuit 3 separates the audio signal into a portion corresponding to the speech sources and a portion corre-

sponding to the audio signal characteristics based on the speech production model under the control of the main controller 8. The pitch detecting circuit 3 processes the separated portions of the audio signal individually.

In order to derive the position of each speech source from the audio signal in this case, a cross-amplitude difference $c(m,\delta)$, which is indicative of a measured signal difference between the (m−1)th speech source and the mth speech source within a tolerance range δ is defined by the following equation (1):

$$c(m,\delta)=\sum_{n=0}^{N-1}|x(n+t(m-1))-x(n+tm+\delta)|$$

where,

x(n): an original input signal (the amount of speech on a time axis n);

tm: the position of the m-th speech source; and

δ: a tolerance region around tm

The cross-amplitude difference is defined as a measure of the similarity between signals by measuring the difference between signals using positions of adjacent speech sources as reference points.

Accordingly, the position of the m-th speech source is determined as the position tm where the cross-amplitude difference is minimized. As this procedure is repeatedly executed for input signals, the speech source components can be extracted.

FIG. 5 shows waveforms of the speech sources and audio characteristics separated in the procedure executed by the analyzing part.

Referring to FIG. 5, it can be seen that general audio signals have substantially similar characteristics in quasi-stationary time intervals, namely, neighboring short time intervals. The longest signal interval involving similar signal characteristics is typically called "one pitch". In the procedure executed by the analyzing part, a pitch interval of the speech source signal is extracted from the input audio signal so that it can be used to adjust the audio play-back speed.

The modulating part executes a procedure for modulating the speech source signal and audio characteristic signal separated in the above-mentioned analyzing procedure. In this regard, the pitch modulating unit 4 includes a speech source modulating unit for processing the speech source signal, and an audio characteristic control unit for executing a smoothing procedure based on the window function needed for a re-synthesis while maintaining the tone color, namely, the audio characteristic.

The speech source modulating unit of the pitch modulating unit 4 deletes or adds the speech source component extracted from the audio signal depending on the play-back speed, thereby adjusting the length of the audio signal. This will be described in more detail in conjunction with FIG. 6.

FIG. 6 illustrates an example of the procedure for modulating the speech sources by the speed-variable audio play-back apparatus shown in FIG. 2.

Where the audio play-back speed is to be decreased, additional speech sources are added while still maintaining the interval of neighboring speech sources, thereby lengthening audio signals. On the other hand, a doubling of the audio play-back speed is achieved by selecting every other speech source while still maintaining the interval of neighboring speech sources and re-synthesizing the selected speech sources using the audio characteristic.

The audio characteristic control unit of the pitch modulating unit 4 performs a signal modulation by applying a

window function, which provides a certain signal length extending from the position of each speech source, to the audio signal portion corresponding to each audio signal characteristic as indicated by the following equation (2):

$$x_m(n)=h_m(t_m-n)x(n) \qquad (2)$$

where,

$x_m(n)$: a modulated audio signal;

$h_m(n)$: the window function;

t_m : the position of each speech source; and

x(n): an input audio signal (the amount of speech on a time axis n).

This procedure produces a smooth audio signal even when a signal modulation has been made by a deletion or addition of speech sources by a speech synthesis that will be described hereinafter.

Finally, the speech source synthesizing circuit 5, which executes a synthesizing procedure, derives a speed-varied speech signal x(n) by utilizing the speech source component and audio signal characteristic modulated in the modulating procedure. The derived speech signal x(n) can be expressed by the following equation (3):

$$x(n)=\frac{\sum_q \alpha q\, xq(n)hq^2(tq-n)}{\sum_q hq^2(tq-n)} \qquad (3)$$

where,

αq: a variable for adjusting the amount of synthesized speech;

xq(n): a modulated audio characteristic ($xq(n)-x_m(n-\delta q)$);

tq: the position of each modulated speech source; and

δq: a variable for determining the play-back speed.

The speed-varied speech signal x(n) is sent to the D/A converter 7 via the output buffer 6. In the D/A converter 7, the speech signal x(n) is converted into an analog signal which is, in turn, output as an audio-out signal.

Where audio is played back using the above system, it can be heard as when a person speaks quickly or slowly even when the play-back speed is varied because the tone color of the speech being played back is maintained.

When videos are monitored or retrieved by high speed play-back in a VTR player, it is possible to obtain a played-back speech while maintaining the original tone color, as when a person speaks quickly or slowly, without causing listeners to feel uncomfortable by variations in tone color or loss of audio signals, both of which occur in existing VTR players.

The present invention is also suitable for high-speed scanning in multimedia equipment. This technique will become more widely used as the growth of the multimedia field continues.

As is apparent from the above description, the present invention provides a speed-variable audio play-back apparatus capable of playing back audio or speech stored in a storage medium at an adjusted speed while preventing any degradation in tone color and loss of audio signals from occurring upon varying the play-back speed while the audio or speech is played back by an audio play-back apparatus such as a tape player, VTR, multimedia equipment, computer and the like, so that the audio (speech) being played back can be heard as when a person speaks quickly or slowly.

Such effects of the present invention are useful in fields associated with design, manufacture and sale of various audio play-back apparatus.

Although the preferred embodiments of the invention have been disclosed for illustrative purposes, those skilled in the art will appreciate that various modifications and additions are possible, without departing from the scope and spirit of the invention as disclosed in the accompanying claims.

What is claimed is:

1. A speed-variable audio play-back apparatus comprising:

a pitch detecting circuit for separating speech source components and audio characteristics from an input audio signal;

a pitch modulating unit for modulating the input audio signal by modulating the separated speech source components and the audio characteristics separated by said pitch detecting circuit, the separated speech source components being modulated by performing one of deleting selected ones of the separated speech source components and adding at least one of the separated speech source components to the separated speech source components, depending on a play-back speed, thereby adjusting a length of an audio signal to be played back;

a speech synthesizing circuit for synthesizing the speech source components modulated by said pitch modulating unit and the audio characteristics modulated by said pitch modulating unit, thereby producing a speed-varied audio signal; and

a main controller for controlling said pitch detecting circuit, said pitch modulating unit, and said speech synthesizing circuit in accordance with control signals externally applied thereto, respectively.

2. The speed-variable audio play-back apparatus of claim 1, wherein the pitch detecting circuit is provided with an analog/digital converter for converting the input audio signal from an analog audio signal to a digital audio signal so that the pitch detecting circuit detects pitch portions of the digital audio signal in a digital manner.

3. The speed-variable audio play-back apparatus of claim 1, wherein the speech synthesizing circuit is provided with a digital/analog converter for converting the speed-varied audio signal into an analog signal.

4. The speed-variable audio play-back apparatus of claim 1, further comprising a memory unit for temporarily storing the input audio signal and sending the stored input audio signal to the speech synthesizing circuit so that the audio signal is compared with the speed-varied audio signal synthesized by the speech synthesizing circuit.

5. The speed-variable audio play-back apparatus of claim 1, further comprising a command memory circuit for storing various control signals required for producing the speed-varied audio signal, receiving control signals from the main controller and outputting the stored control signals respectively based on the received control signals.

6. The speed-variable audio play-back apparatus of claim 1, wherein the pitch detecting circuit separates the speech source components on the basis of the following equation:

$$c(m,\delta)=\sum_{n=0}^{N-1}|x(n+t(m-1))-x(n+tm+\delta)|$$

where,

x(n): the input audio signal (an amount of speech on a time axis n);

tm: a position of an m-th speech source;

δ: a tolerance region around tm;

c(m,δ): a cross-amplitude difference.

7. The speed-variable audio play-back apparatus of claim 1, wherein the pitch modulating unit modulates the input audio signal by applying a window function which provides a required signal length extending from a position of each separated speech source component to a portion of the input audio signal corresponding to each audio signal characteristic as expressed by the following equation:

$$x_m(n)=h_m(t_m-n)x(n)$$

where,

$x_m(n)$: the modulated input audio signal;

$h_m(n)$: the window function;

t_m: the position of each separated speech source is component; and

x(n): the input audio signal (an amount of speech on a time axis n).

8. The speed-variable audio play-back apparatus of claim 1, wherein the speech synthesizing circuit derives the speed-varied audio signal by use of the modulated separated speech source components and the modulated audio signal characteristics as expressed by the following equation:

$$x(n)=\frac{\sum_q \alpha q x q(n) h q^2(tq-n)}{\sum_q hq^2(tq-n)}$$

where,

x(n): the speed-varied audio signal;

αq: a variable for adjusting an amount of synthesized speech;

xq(n): the modulated audio characteristics ($xq(n)=x_m(n-\delta q)$);

tq: a position of each modulated separated speech source; and

δq: a variable for determining play-back speed.

* * * * *

US006915263B1

(12) **United States Patent**
Chen et al.

(10) **Patent No.: US 6,915,263 B1**
(45) **Date of Patent: Jul. 5, 2005**

(54) **DIGITAL AUDIO DECODER HAVING ERROR CONCEALMENT USING A DYNAMIC RECOVERY DELAY AND FRAME REPEATING AND ALSO HAVING FAST AUDIO MUTING CAPABILITIES**

(75) Inventors: **Hua Chen**, San Jose, CA (US); **Ikuo Tsukagoshi**, Sunnyvale, CA (US); **Milan Mehta**, Fremont, CA (US)

(73) Assignees: **Sony Corporation**, Tokyo (JP); **Sony Electronics, Inc.**, Park Ridge, NJ (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/422,134**

(22) Filed: **Oct. 20, 1999**

(51) **Int. Cl.**[7] **G10L 19/00**

(52) **U.S. Cl.** **704/500**; 704/226

(58) **Field of Search** 704/226, 500, 704/228, 230, 229, 201, 265, 212, 206, 223; 364/715.02

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,271,011 A * 12/1993 McMullan et al. 714/807
5,309,443 A 5/1994 Schorman 371/5.5
5,347,478 A * 9/1994 Suzuki et al. 708/203
5,533,133 A 7/1996 Lamkin et al. 381/94
5,890,112 A * 3/1999 Kitabatake 704/229
5,918,205 A * 6/1999 Dierke 704/230
5,943,347 A 8/1999 Shepard 371/31
5,983,174 A * 11/1999 Wong et al. 704/228
6,009,389 A * 12/1999 Dokic et al. 704/228
6,687,670 B2 * 2/2004 Sydanmaa et al. 704/226
2002/0082827 A1 * 6/2002 Wiese et al. 704/201
2002/0147590 A1 * 10/2002 Sydanmaa et al. 704/265

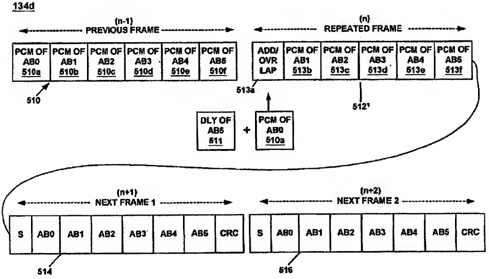
* cited by examiner

Primary Examiner—David L. Ometz
Assistant Examiner—Jakieda Jackson
(74) *Attorney, Agent, or Firm*—Wagner, Murabito & Hao LLP

(57) **ABSTRACT**

A multimedia decoder unit having error concealment and fast muting capabilities. The audio decoder provides error concealment using a dynamic recovery delay that is based on the error rate of an input digital bitstream and also uses frame repeating. The decoder allows fast audio muting whereby audio can be muted within two audio frames of a mute signal that immediately freezes the video frame, e.g., a channel change. With respect to the dynamic recovery delay, a template of fixed length is used to inspect the last frames within the template. If error is found, then the error sum is used as an index into a table length which provides a dynamic template length. Error within the dynamic template length is computed and if larger than a tolerance, the current frame is muted. This allows the recovery delay to be adaptive and based on the error rate while still allowing mute merging. Muting the current frame can be achieved by repeating the previous frame but the delay data of the last block of the previous audio frame is added to the first block of the repeated audio frame to provide a smooth frame interface. In response to a mute command, the decoder zeros the audio output bitstream stream to provide zero frames at the audio output buffer (AOB). In addition, the decoder also directly zeros audio frames in the AOB that lie between its the read and write pointers to guarantee that only two frames of audio be played after the mute signal.

14 Claims, 18 Drawing Sheets



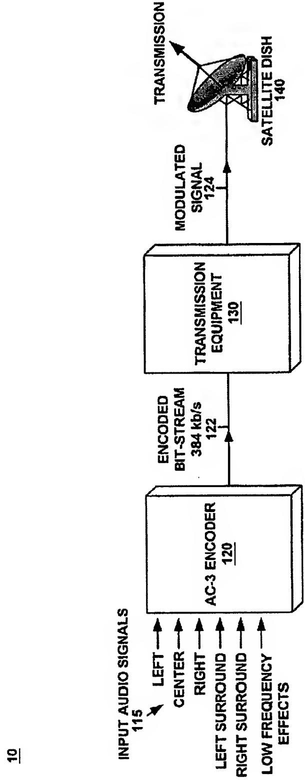

10
INPUT AUDIO SIGNALS
115
LEFT
CENTER
RIGHT
LEFT SURROUND
RIGHT SURROUND
LOW FREQUENCY EFFECTS
AC-3 ENCODER
120
ENCODED
BIT-STREAM
384 kb/s
122
TRANSMISSION EQUIPMENT
130
MODULATED SIGNAL
124
TRANSMISSION
SATELLITE DISH
140

FIGURE 1A

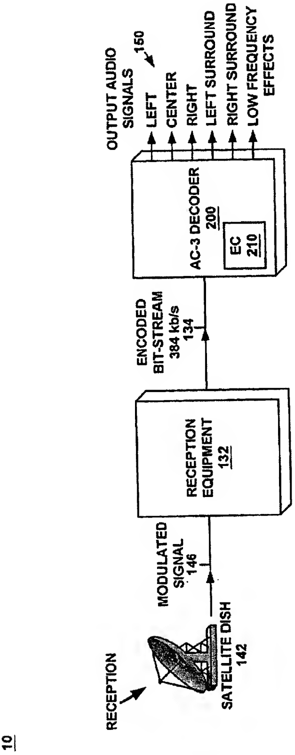
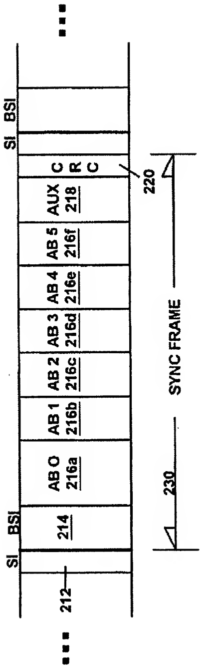

10
RECEPTION
SATELLITE DISH
142
MODULATED
SIGNAL
146
RECEPTION
EQUIPMENT
132
ENCODED
BIT-STREAM
384 kb/s
134
AC-3 DECODER
200
EC
210
OUTPUT AUDIO
SIGNALS
150
LEFT
CENTER
RIGHT
LEFT SURROUND
RIGHT SURROUND
LOW FREQUENCY
EFFECTS

FIGURE 1B


212
SI
BSI
214
AB O
216a
AB 1
216b
AB 2
216c
AB 3
216d
AB 4
216e
AB 5
216f
AUX
218
C
R
C
220
SI
BSI
230
SYNC FRAME

134

FIGURE 2

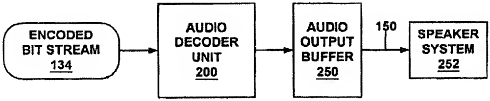

ENCODED
BIT STREAM
134
AUDIO
DECODER
UNIT
200
AUDIO
OUTPUT
BUFFER
250
150
SPEAKER
SYSTEM
252

FIGURE 3A

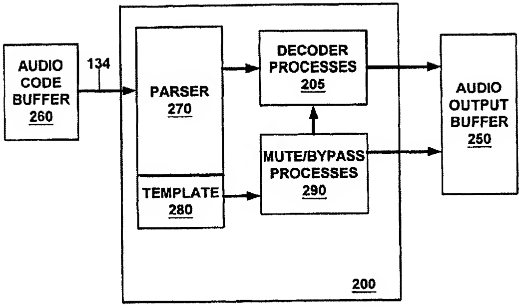

AUDIO
CODE
BUFFER
260
134
PARSER
270
TEMPLATE
280
DECODER
PROCESSES
205
MUTE/BYPASS
PROCESSES
290
200
AUDIO
OUTPUT
BUFFER
250

FIGURE 3B

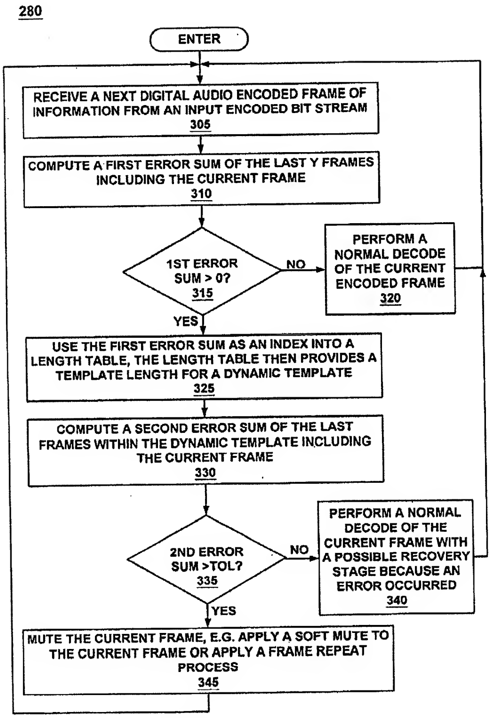

280
ENTER
RECEIVE A NEXT DIGITAL AUDIO ENCODED FRAME OF INFORMATION FROM AN INPUT ENCODED BIT STREAM 305
COMPUTE A FIRST ERROR SUM OF THE LAST Y FRAMES INCLUDING THE CURRENT FRAME 310
1ST ERROR SUM > 0? 315
NO
PERFORM A NORMAL DECODE OF THE CURRENT ENCODED FRAME 320
YES
USE THE FIRST ERROR SUM AS AN INDEX INTO A LENGTH TABLE, THE LENGTH TABLE THEN PROVIDES A TEMPLATE LENGTH FOR A DYNAMIC TEMPLATE 325
COMPUTE A SECOND ERROR SUM OF THE LAST FRAMES WITHIN THE DYNAMIC TEMPLATE INCLUDING THE CURRENT FRAME 330
2ND ERROR SUM >TOL? 335
NO
PERFORM A NORMAL DECODE OF THE CURRENT FRAME WITH A POSSIBLE RECOVERY STAGE BECAUSE AN ERROR OCCURRED 340
YES
MUTE THE CURRENT FRAME, E.G. APPLY A SOFT MUTE TO THE CURRENT FRAME OR APPLY A FRAME REPEAT PROCESS 345

FIGURE 4

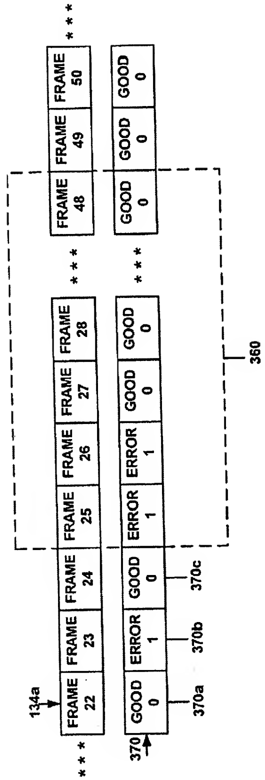

134a
FRAME 22
FRAME 23
FRAME 24
FRAME 25
FRAME 26
FRAME 27
FRAME 28
FRAME 48
FRAME 49
FRAME 50
370
GOOD 0
ERROR 1
GOOD 0
ERROR 1
ERROR 1
GOOD 0
GOOD 0
GOOD 0
GOOD 0
GOOD 0
370a
370b
370c
360

FIGURE 5A

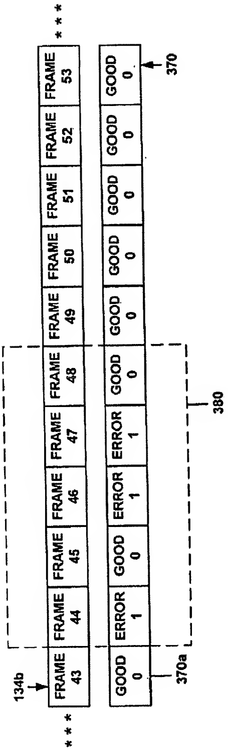

134b
FRAME 43
FRAME 44
FRAME 45
FRAME 46
FRAME 47
FRAME 48
FRAME 49
FRAME 50
FRAME 51
FRAME 52
FRAME 53
GOOD 0
ERROR 1
GOOD 0
ERROR 1
ERROR 1
GOOD 0
GOOD 0
GOOD 0
GOOD 0
GOOD 0
GOOD 0
370a
380
370

FIGURE 5B

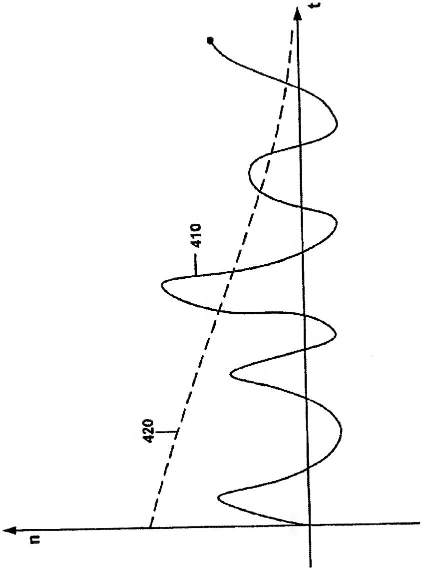


FIGURE 6

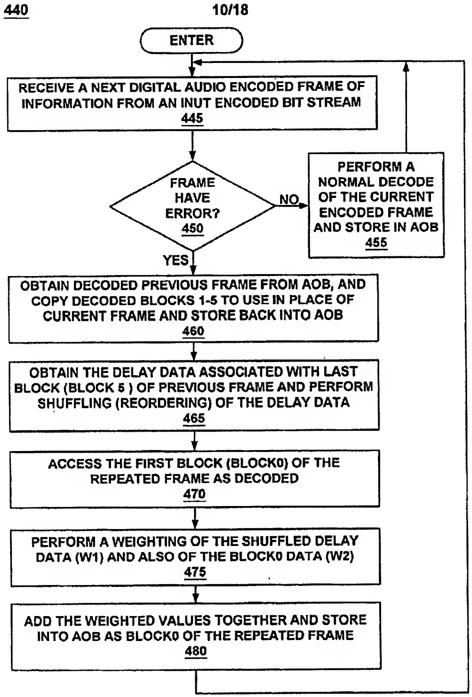

440
ENTER
RECEIVE A NEXT DIGITAL AUDIO ENCODED FRAME OF INFORMATION FROM AN INUT ENCODED BIT STREAM 445
FRAME HAVE ERROR? 450
NO
PERFORM A NORMAL DECODE OF THE CURRENT ENCODED FRAME AND STORE IN AOB 455
YES
OBTAIN DECODED PREVIOUS FRAME FROM AOB, AND COPY DECODED BLOCKS 1-5 TO USE IN PLACE OF CURRENT FRAME AND STORE BACK INTO AOB 460
OBTAIN THE DELAY DATA ASSOCIATED WITH LAST BLOCK (BLOCK 5) OF PREVIOUS FRAME AND PERFORM SHUFFLING (REORDERING) OF THE DELAY DATA 465
ACCESS THE FIRST BLOCK (BLOCK0) OF THE REPEATED FRAME AS DECODED 470
PERFORM A WEIGHTING OF THE SHUFFLED DELAY DATA (W1) AND ALSO OF THE BLOCK0 DATA (W2) 475
ADD THE WEIGHTED VALUES TOGETHER AND STORE INTO AOB AS BLOCK0 OF THE REPEATED FRAME 480

FIGURE 7

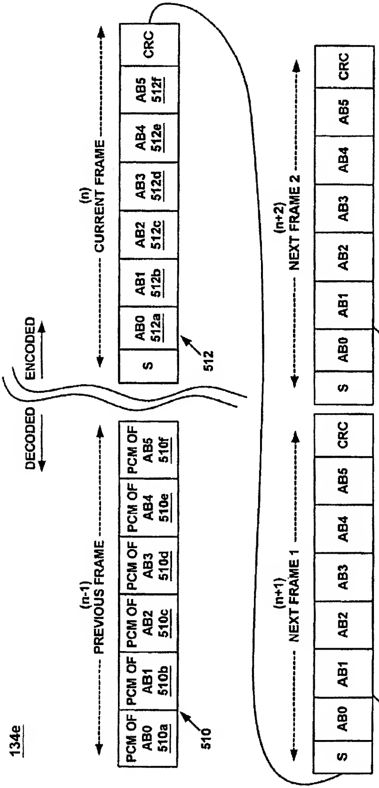

134e
DECODED
ENCODED
(n-1)
PREVIOUS FRAME
PCM OF AB0 510a
PCM OF AB1 510b
PCM OF AB2 510c
PCM OF AB3 510d
PCM OF AB4 510e
PCM OF AB5 510f
510
(n)
CURRENT FRAME
S
AB0 512a
AB1 512b
AB2 512c
AB3 512d
AB4 512e
AB5 512f
CRC
512
(n+1)
NEXT FRAME 1
S
AB0
AB1
AB2
AB3
AB4
AB5
CRC
514
(n+2)
NEXT FRAME 2
S
AB0
AB1
AB2
AB3
AB4
AB5
CRC
516

FIGURE 8

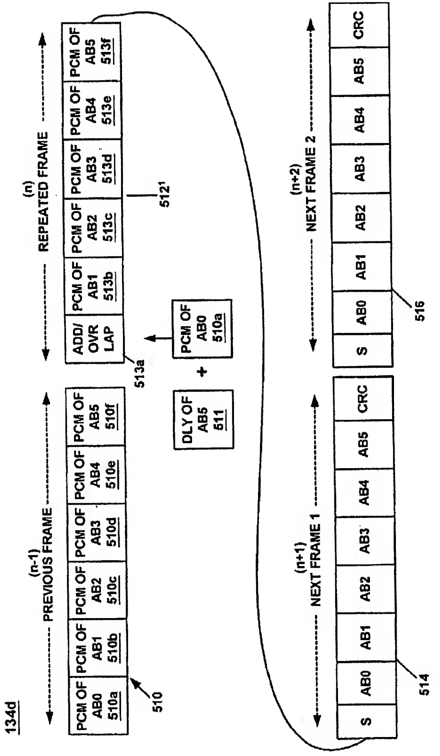

134d
(n-1)
PREVIOUS FRAME
PCM OF AB0 510a
PCM OF AB1 510b
PCM OF AB2 510c
PCM OF AB3 510d
PCM OF AB4 510e
PCM OF AB5 510f
510
(n)
REPEATED FRAME
ADD/ OVR LAP
513a
PCM OF AB1 513b
PCM OF AB2 513c
PCM OF AB3 513d
PCM OF AB4 513e
PCM OF AB5 513f
512'
DLY OF AB5 511
+
PCM OF AB0 510a
(n+1)
NEXT FRAME 1
S
AB0
AB1
AB2
AB3
AB4
AB5
CRC
514
(n+2)
NEXT FRAME 2
S
AB0
AB1
AB2
AB3
AB4
AB5
CRC
516

FIGURE 9

600

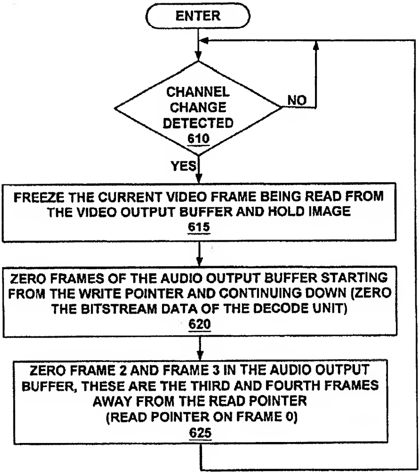


FIGURE 10

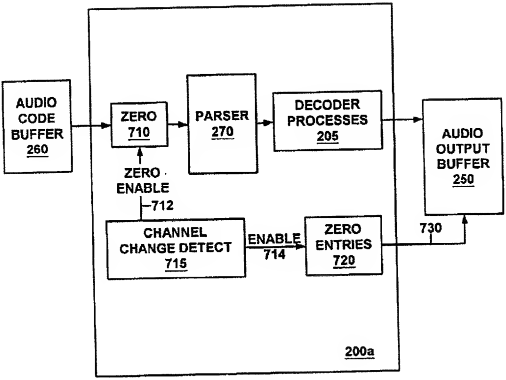

AUDIO CODE BUFFER 260
ZERO 710
PARSER 270
DECODER PROCESSES 205
AUDIO OUTPUT BUFFER 250
ZERO ENABLE
712
CHANNEL CHANGE DETECT 715
ENABLE 714
ZERO ENTRIES 720
730
200a

FIGURE 11

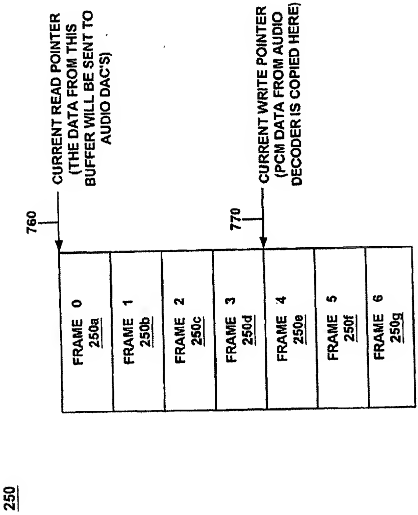

250
760
CURRENT READ POINTER (THE DATA FROM THIS BUFFER WILL BE SENT TO AUDIO DAC'S)
770
CURRENT WRITE POINTER (PCM DATA FROM AUDIO DECODER IS COPIED HERE)
FRAME 0 250a
FRAME 1 250b
FRAME 2 250c
FRAME 3 250d
FRAME 4 250e
FRAME 5 250f
FRAME 6 250g

FIGURE 12

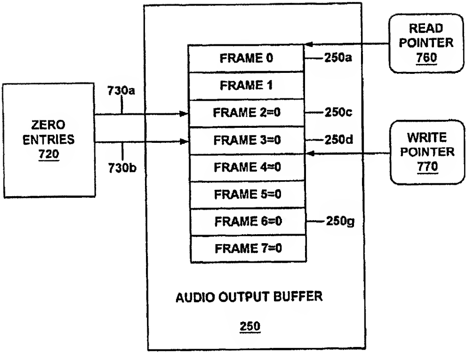

READ POINTER 760
FRAME 0
250a
FRAME 1
730a
ZERO ENTRIES 720
FRAME 2=0
250c
FRAME 3=0
250d
WRITE POINTER 770
730b
FRAME 4=0
FRAME 5=0
FRAME 6=0
250g
FRAME 7=0
AUDIO OUTPUT BUFFER
250

FIGURE 13

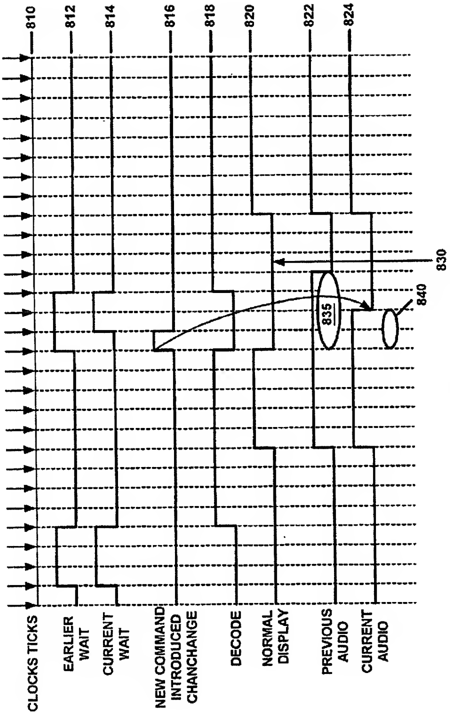
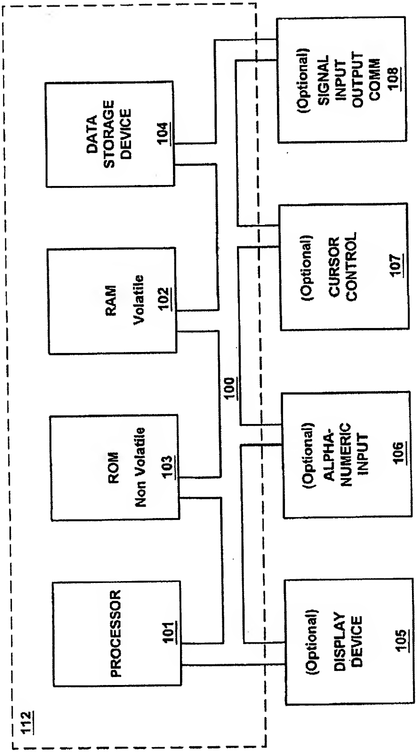

810
812
814
816
818
820
822
824
830
835
840
CLOCKS TICKS
EARLIER WAIT
CURRENT WAIT
NEW COMMAND INTRODUCED CHANCHANGE
DECODE
NORMAL DISPLAY
PREVIOUS AUDIO
CURRENT AUDIO

FIGURE 14



DATA STORAGE DEVICE
104
RAM
Volatile
102
ROM
Non Volatile
103
PROCESSOR
101
100
112
(Optional)
SIGNAL INPUT OUTPUT COMM
108
(Optional)
CURSOR CONTROL
107
(Optional)
ALPHA-NUMERIC INPUT
106
(Optional)
DISPLAY DEVICE
105

FIGURE 15

DIGITAL AUDIO DECODER HAVING ERROR CONCEALMENT USING A DYNAMIC RECOVERY DELAY AND FRAME REPEATING AND ALSO HAVING FAST AUDIO MUTING CAPABILITIES

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to the field of multimedia electronic systems. More particularly, the present invention relates to an audio decoder unit for decoding digital multimedia bitstreams representing audio information.

2. Related Art

Audio/visual (AV) material is increasingly stored, transmitted and rendered using digital data. Digital video representation of AV material facilitates its usage with computer controlled electronics and also facilitates high quality image and sound reproduction. Digital AV material is typically compressed ("encoded") in order to reduce the computer resources required to store and transmit the digital data. The systems that transmit multimedia content encode and/or compress the content to use their transmission channel efficiently because the size of the multimedia content, especially video, is very large. For instance, in order to more efficiently broadcast or record audio signals, the amount of information required to represent the audio signals can be reduced. In the case of digital audio signals, the amount of digital information needed to accurately reproduce the original pulse code modulation (PCM) samples can be reduced by applying a digital compression process, such as AC3, for instance, resulting in a digitally compressed representation of the original sample.

Digital AV material can be encoded using a number of well known standards including, for example, the AC3 audio standard, the DV (Digital Video) standard, the MPEG (Motion Picture Expert Group) standard, the JPEG standard, the H.261 standard, the H.263 standard and the Motion JPEG standard to name a few. The encoding standards also specify the associated decoding processes as well. The multimedia contents are typically stored on the storage media and are transmitted as bitstreams which represent audio for video frames. In particular, the ATSC digital terrestrial transmission standard adopts the AC3 format for audio encoding and the MPEG2 format for video encoding.

MPEG is the compression standard for audio, video and graphics information and includes, for example, MPEG1, 2, 4 and 7. It is standardized in the ISO-IEC/JTC1/SC29/WG11 documents. MPEG1 is the standard for encoding audio and video data for storage on CD-ROM devices (compact disc read only memory). The MPEG1 specification is described in the IS-11393 standard. MPEG2 is the standard (adopted for ATSC) for encoding, decoding and transmitting video data for storage media, e.g., DVD (digital video disc), etc., and also for digital broadcasts. MPEG2 supports interlaced video while MPEG1 does not. Therefore, MPEG2 is used for high quality video displaying on TV units. The MPEG2 specification is described in IS-13818. The MPEG4 standard is used for encoding, decoding and transmitting audio, video and computer graphics data. It supports content based bitstream manipulation and representation. The specification is described in IS14496. MPEG7 is the standard of the meta information of multimedia (MM) contents. The example of the meta data is data is describes or is related to the MM contents, such as, identification and/or other descriptions of the author, producer information, directors, actors, etc. The MPEG7 standard is currently under standardization, and is in draft form but available. The draft specifications are described in the ISO-IEC-JTC1/SC29/WG11 documents.

One problem with using encoded digital audio information is that errors can occur between the transmission and reception of the audio data. The decoder unit can detect when a particular frame of the audio data contains error by using well known CRC checking schemes. In the past, the frame having the error would be muted by filling in the frame with zeros. This is called a hard mute. However, the hard mute, when played back, causes a very audible "pop" sound which is not pleasing to the ear nor does it sound natural. Therefore, an attenuation function or "window" was applied to the error frame to soften the mute. However, even soft mutes can have a "pop" associated therewith depending on the window function applied. Also, hard and soft mutes still have a duration of silence associated therewith that can be distinguished by the ear. Therefore, when many error frames are detected in the same bitstream neighborhood, these intermittent durations of silence (mutes) followed by sound (unmute) and silence again (mute) can be very unappealing to the ear and annoying and can also damage speaker systems.

Another problem with using encoding digital audio information involves muting commands and audio signal synchronization. For instance, if a user watching a program on a digital TV changes the current channel, the currently played AV information should stop incident to the channel change, e.g., mute audio and freeze the video, then the channel should change. However, in conventional systems, audio artifacts can result because the audio may not mute fast enough as a result of situations described below. The video signal is used as a master and the audio signal is the slave in many encoding schemes. Also, the amount of playback time in a video frame may not be exactly the same as in a video frame in many encoding standards. Therefore, the audio frames and video frames are not exactly synchronized in the decoding and playback processes. Secondly, the channel change operation takes some time to complete because the AV system needs to parse the bitstream from the new channel and feed the data to corresponding audio and video decoders. This results in a situation where the audio and video are slightly delayed during decoding and playback. When the decoder receives a mute command, it is able to immediately freeze the video frame, because the video signal is the master. However, many decoded audio frames may be stored in the output buffer, resulting in some audio playback after the video freeze. This is very noticeable to the ear and confusing because the audio playback coincides with video frames that are not displayed simultaneously.

SUMMARY OF THE INVENTION

Accordingly, the present invention provides an audio decoder unit that merges nearby muted ("error") frames to extend a silence period between the error frames when the error rate is high. By extending the silence period, a more natural and less annoying sound results when the bitstream includes many nearby errors. This mute merging can be accomplished using a dynamically adjusted recovery delay period that is adaptive based on the error rate. By extending the recovery period, mutes are merged, e.g., non-error frames are muted to provide a longer mute duration. The present invention also applies a novel frame repeating technique for frame muting to conceal single silent frame periods without "pops" or other audio artifacts that result during single frame muting. In addition, the present inven-

tion provides an effective mechanism for guaranteeing that only two audio frames are played back incident to a mute command. This provides a better sounding channel change.

In one embodiment, the present invention provides a multimedia information decoder unit having error concealment and fast muting capabilities. The audio decoder provides error concealment using a dynamic recovery delay that is based on the error rate of an input digital bitstream and also uses frame repeating. The decoder also allows fast audio muting whereby audio can be muted within two audio frames of a mute signal that immediately freezes the video frame, e.g., for use in a channel change situation. With respect to the dynamic recovery delay, a template of fixed length, e.g., 24 audio frames, is used to inspect the last frames within the template. If error is found in this fixed template, then the error sum is used as an index into a table length which provides a dynamic template length. Error within the dynamic template length is then computed and if larger than a prescribed tolerance, the current frame is muted. This allows the recovery delay to be adaptive and based on the error rate while still allowing mute merging. When muting is performed, smoothed muting can be used in one embodiment and in another embodiment, frame repeating can be performed.

In cases when only one bad frame appears within a neighborhood of otherwise good frames, a single frame mute can be performed. In accordance with the present invention, muting the current frame can also be achieved by repeating, in the time domain, the previous frame. In single frame muting cases, the delay data of the last block of the previous audio frame is added to the first block of the repeated audio frame to provide a smooth interface between the repeated frame. Before the addition, data reordering and weighting are performed. In response to a mute command (e.g., incident to a channel change), the decoder zeros the audio output bitstream stream to provide zero frames at the audio output buffer (AOB). In addition, the decoder also directly zeros audio frames in the AOB that lie between its the read and write pointers to guarantee that only two frames of audio be played after the mute signal.

More specifically, A first embodiment of the present invention is drawn to a method for muting a portion of an encoded bitstream of audio information comprising the steps of: a) with respect to a current encoded audio frame of the encoded bitstream, computing a length of a dynamic template based on an error rate of the encoded bitstream, the dynamic template encompassing a plurality of previous encoded frames of the encoded bitstream; b) summing errors of the plurality of previous encoded frames within the dynamic template to produce a first error sum; c) determining if the first error sum exceeds a prescribed tolerance; and d) adaptively merging muted error frames by muting the current encoded audio frame provided the first error sum value exceeds the prescribed tolerance whether or not the current encoded audio frame has an error. A variation of the first embodiment further includes a method as described above wherein the step a) comprises the steps of: a1) with respect to the current encoded audio frame, summing errors of a plurality of previous encoded frames encompassed by a fixed-length template to produce a second error sum; and a2) using the second error sum as an index to a look-up table to compute the length of the dynamic template.

A second embodiment of the present invention includes a method for muting a portion of an encoded bitstream of audio information comprising the steps of: a) detecting if a current encoded audio frame of the encoded bitstream contains an error; and b) provided an error is detected, repeating a previous decoded audio frame in lieu of the current encoded audio frame, the step b) comprising the steps of: b1) obtaining decoded data of the previous audio frame; b2) generating a repeated audio frame by replicating the decoded data of the previous audio frame for use in lieu of the current encoded audio frame; b3) modifying the repeated audio frame by adding delay information of a last block of the previous audio frame with pulse code modulated (PCM) data of a first block of the repeated audio frame to generate new decoded data for the first block of the repeated audio frame; and b4) sending the repeated audio frame to an audio output buffer for playout.

A third embodiment of the present invention includes a method (within a digital decoder) for reducing audio frame over-run comprising the steps of: a) responsive to an audio mute signal, causing an input audio encoded bitstream to zero, the step a) causing entries in an audio output buffer to zero starting from an entry position pointed to by a write pointer associated with the audio output buffer; and b) directly zeroing a plurality of entries of the audio output buffer in response to the audio mute signal, the plurality of entries being a few entries away from a read pointer of the audio output buffer, the read pointer following the write pointer and wherein as a result of step a) and step b), only a predetermined number of audio output frames are guaranteed to be played after the audio mute signal is detected.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1A and FIG. 1B illustrate an exemplary multimedia communication system including a transmission system having an encoder and a reception system having a decoder unit.

FIG. 2 illustrates an exemplary encoded audio frame of an encoded digital audio bitstream.

FIG. 3A is a block diagram of an audio decoding system in accordance with the present invention.

FIG. 3B is a block diagram of an audio decoding system having an error concealment circuit in accordance with one aspect of the present invention.

FIG. 4 illustrates steps in a process in accordance with one embodiment of the present invention for providing a dynamic error recovery delay with mute merging.

FIG. 5A illustrates a portion of the encoded digital audio stream and the fixed template used in accordance with the embodiment of the present invention shown in FIG. 4.

FIG. 5B illustrates a portion of the encoded digital audio stream and the dynamic template used in accordance with the embodiment of the present invention of FIG. 4.

FIG. 6 illustrates an audio signal in accordance with a muted audio frame and a smoothing window function applied thereto.

FIG. 7 illustrates steps in a process in accordance with an embodiment of the present invention for perform frame repeating for a single frame muting operation.

FIG. 8 illustrates a portion of the encoded audio bitstream having a single frame to be muted in accordance with the embodiment of the present invention of FIG. 7.

FIG. 9 illustrates the portion of the encoded audio bitstream of FIG. 8 after frame repeating in accordance with the embodiment of the present invention of FIG. 7.

FIG. 10 illustrates steps in a process in accordance with an embodiment of the present invention for reducing the number of audio frames played back following an audio mute command.

FIG. 11 is a block diagram of a decoder unit in accordance with the embodiment of the present invention of FIG. 10.

FIG. 12 illustrates the contents of the audio output buffer of an audio decoder system.

FIG. 13 illustrates the contents of the audio output buffer of an audio decoder system after frame zeroing in accordance with the embodiment of the present invention of FIG. 10.

FIG. 14 illustrates a timing diagram of signals involved in the embodiment of the present invention of FIG. 10.

FIG. 15 is a block diagram of a computer system platform on which the error concealment and muting embodiments of the present invention can be practiced.

DETAILED DESCRIPTION OF THE INVENTION

In the following detailed description of the present invention, a digital audio decoder system for a multimedia information system having improved error concealment functionality, improved muting capabilities and reduced audio frame overrun in response to a mute-command, numerous specific details are set forth in order to provide a thorough understanding of the present invention. However, it will be recognized by one skilled in the art that the present invention may be practiced without these specific details or with equivalents thereof. In other instances, well known methods, procedures, components, and circuits have not been described in detail as not to unnecessarily obscure aspects of the present invention.

Notation and Nomenclature

Some portions of the detailed descriptions which follow are presented in terms of procedures, steps, logic blocks, processing, and other symbolic representations of operations on data bits that can be performed on computer memory. These descriptions and representations are the means used by those skilled in the data processing arts to most effectively convey the substance of their work to others skilled in the art. A procedure, computer executed step, logic block, process, etc., is here, and generally, conceived to be a self-consistent sequence of steps or instructions leading to a desired result. The steps are those requiring physical manipulations of physical quantities. Usually, though not necessarily, these quantities take the form of electrical or magnetic signals capable of being stored, transferred, combined, compared, and otherwise manipulated in a computer system. It has proven convenient at times, principally for reasons of common usage, to refer to these signals as bits, values, elements, symbols, characters, terms, numbers, or the like.

It should be borne in mind, however, that all of these and similar terms are to be associated with the appropriate physical quantities and are merely convenient labels applied to these quantities. Unless specifically stated otherwise as apparent from the following discussions, it is appreciated that throughout the present invention, discussions utilizing terms such as "processing" or "computing" or "translating" or "calculating" or "determining" or "scrolling" or "displaying" or "recognizing" or the like, refer to the action and processes of a computer system, or similar electronic computing device, that manipulates and transforms data represented as physical (electronic) quantities within the computer system's registers and memories into other data similarly represented as physical quantities within the computer system memories or registers or other such information storage, transmission or display devices.

Audio Decoder System

Embodiments of the present invention are directed to a digital audio decoder system **200** as shown in FIG. 1B. FIG. 1A and FIG. 1B together illustrate a multimedia communication system **10**. In accordance with the system **10**, a multimedia encoder system **120** accepts an audio signal (e.g., PCM audio) **115** and produces an encoded bitstream **122** based thereon. In one embodiment, this is an AC3 format compliant encoded signal which has a frequency of 384 kilobytes per second. This encoded bitstream is processed by transmission equipment **130** to produce a modulated signal **124** which can be transmitted **140**, e.g., by a satellite dish or other suitable cable broadcast system, etc.

FIG. 1B illustrates the receiving system which includes receiver hardware **142** (satellite dish, cable receiver, etc.) and reception equipment **132** capable of converting the received modulated signal **146** to an encoded digital bitstream **134**. It is appreciated that encoded bitstream **134** may vary from encoded bitstream **122** as a result of one or more signal errors that can be a result of transmission/reception problems. The encoded bitstream **134** is then fed to a digital decoder unit **200** which generates an output signal **150** that can be fed to a speaker system for rendering audible signals. The decoder system **200** of the present invention contains error concealment circuitry **210** for processing audio frames that have signal errors therein.

FIG. 2 illustrates a frame **230** of the encoded audio signal **134**. An AC3 serial coded audio bitstream **134** is made up of a sequence of audio synchronization frames ("frames"). Each frame consists of **6** coded audio blocks (AB) **216*a*–216*f*** each of which represent encoded data of **256** new audio samples. The samples are made at 48 kHz. When decoded, each frame **230** represents **32** ms of playback time. A synchronization information (SI) header **212** is located at the start of each frame **230** and contains information needed to acquire and maintain synchronization. A bitstream information (BSI) header **214** follows SI header **212**, and contains parameters describing the coded audio service. The coded audio blocks **216** can be followed by an auxiliary data field **218**. At the end of each frame is an error check field that includes a CRC (cyclic redundancy check) word **220** for error detection. An additional CRC word located in the SI header **212** is optional and can be included within each frame **230**.

FIG. 3A is a logical block diagram of an decoder system in accordance with an embodiment of the present invention. The decoder unit **200** receives an encoded audio bitstream **134** and forwards decoded audio frames to an audio output buffer AOB **250**. The AOB **250** contains several decoded frames, some of which are required as a result of audio-video delay. A read pointer marks the memory position at which audio frames are removed from the AOB **250** and sent over line **150** to the speaker system **252**. A write pointer marks the memory position where new audio frames are received from the decoder unit **200**. A circular buffer management technique can be used in the AOB **250**.

FIG. 3B illustrates a more detailed view of the decoder unit **200**. Decoder unit **200** contains a parser **270**, template processing unit **280**, a decoder processing unit **205** and a mute/bypass processing unit **290**. It is appreciated that the components of the decoder unit **200** can be realized using hardware circuitry or can be realized using software. The decoder processing unit **205** and the mute/bypass processing unit **290** both are coupled to the AOB **250**. The parser **270** scans the input encoded bitstream **134** (which can originate from an audio code buffer **260**). In one embodiment, the input bitstream **134** is compliant with the ATSC standard which includes an AC3 encoded bitstream for audio information. The template processing unit **280** determines whether or not a current frame is to be muted and therefore

is an error concealment circuit. Template processing unit **280** functions in accordance with the steps of FIG. **4**. If a particular frame is to be muted, that frame is called an error or mute frame, and is processed by the mute/bypass processing unit **290**.

Adaptive Merging of Error Frames Based on Error Rate

FIG. **4** illustrates steps in a process **280** in accordance with one embodiment of the present invention for providing a dynamic error recovery delay with mute merging. Process **280** can be realized in hardware or it can be realized in software. As software, process **280** is realized as instruction code executed by system **112**. (FIG. **15**). Process **280** operates by muting some non-error audio frames in order to merge two or more error frames into one longer silence period. This reduces the amount of annoying intermittent silence periods followed by sound and silence again in cases when the error rate is high. In this embodiment of the present invention, the length of the recovery delay is adaptive and depends on the amount of accumulated errors found in the input bitstream **134** (FIG. 3B).

At step **305**, a digital audio encoded frame is received by the decoder **200** from an input bitstream **134**. An exemplary input bitstream **134***a* is shown in FIG. 5A and includes encoded frames **22–50**. It is appreciated that each encoded frame also includes a corresponding array entry of error array **370**. The individual entries of error array **370** are one bit in length and specify whether or not the encoded frame associated with the entry contains an error. In one embodiment, a "1" indicates an error and a "0" indicates no error. For instance, entry **370***a* corresponds to frame **22** and indicates a good frame. Entry **370***b* corresponds to frame **23** and indicates an error frame while entry **370***c* corresponds to frame **24** and indicates a good frame.

The error entries of error array **370** can be computed and stored by the parser process **270** of the decoder **200**. There are several ways in which the AC3 data can indicate that errors are contained within a frame of encoded data. In one method, the decoder **200** can be informed of the error frame by the transport system which delivers the data. The data integrity can also be checked using the embedded CRC **220** fields for each encoded frame. Methods for using the CRC fields of an encoded frame for error detection are well known. Also, well known consistency checks on the received bitstream **134** can also be used to indicate that errors are present in a particular encoded frame. It is appreciated that at step **305** of FIG. **4**, any of a number of well known processes can be used for generating the error array **370** of FIG. 5A based on the input bitstream **134**. In the example of FIG. 5A, the next audio encoded frame that is being processed at step **305** is frame **48**. All other frames of lesser frame number to frame **48** have already been processed by step **305** and are therefore previous frames.

At step **310** of FIG. **4**, a first error sum value (sum_error1) is computed based on the error array entries of the last previous Y frames that were processed by step **305** including the current frame (e.g., frame **48**). In one embodiment, the value of Y is a constant and can be selected based on a number of different considerations. In one implementation, Y=24. Using this example, the first error sum value is therefore computed based on the error entries of the error array **370** for frames **25–48**. A first error template **360** is shown in FIG. 5A and includes the error entries of the last 24 frames that were processed by step **305**. The first error template is called the static or fixed error template because its frame number is constant. The first error sum value is therefore the summation of all error entries that lie within the error template **360**. It is appreciated that the first error template **360** moves along with the decoding process as new frames are processed by step **305**. That is so say, the frames contained in the first error template **360** become updated when a new frame is processed by step **305**. For instance, when frame **49** is the next processed frame, the frames of the error template **360** will include frames **26–49** and so on. It is appreciated that if the current frame contains an error therein, then the first error sum value (sum_error1) will always be greater than zero because the current frame is always included within the first error template **360**.

At step **315**, if the first error sum (sum_error1) is greater than zero, then step **325** is entered otherwise step **320** is entered. At step **320**, no error was detected in the first template **360**, therefore no muting operations are required and normal decoding can occur on the current frame. At step **320**, a normal decode process is performed on the current frame (e.g., frame **48**). In other words, no muting functions are applied to the current frame and decoding processes **205** (FIG. 3B) are applied to the current frame. After the decoding processes, the decoded frame is placed into the audio output buffer **250** at the position of the write pointer and eventually played out. Process **280** then returns to step **305** to obtain and process the next encoded frame.

At step **325**, errors are detected in the first template **360** and muting operations need to be executed. At step **325**, the value of the first error sum (sum_error1) is used as an index into a lookup table called the "length table." Although a variety of different length tables can be used, one exemplary length table is shown below:

int lengthtab[sum_error1]=(1, 1, 1, 1, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1)

It is appreciated that the first three entries of the exemplary length table do not map to **23**, **22** and **21** to avoid very long recovery delays.

The length table provides a length for a second error template that is dynamic in size and depends on the error rate of the input bitstream **134** as determined by the sum_error1 value. FIG. 5B illustrates an exemplary portion **134***b* of the input bitstream and also illustrates an example of the second error template **380** that spans from the current frame (frame **48**) and has a length determined by the above length table. In this example, the length of the second error template is five frames long and includes previous frames **44–47** and the current frame **48**. The second error template is called the dynamic or adaptive template because its length is not fixed but varies based on the error rate of the input bitstream.

At step **330** of FIG. **4**, a second error sum value (sum_error) is then computed based on the summation of the error entries of the error array **370** for the frames of the second error template **380**. In this case, the second error sum value is 1+0+1+1+0 or 5. It is appreciated that if the current frame contains an error therein, then the second error sum value (sum_error2) will always be greater than zero because the current frame is always included within the second error template **380**. At step **335**, a check is made to determine if the second error sum is greater than a prescribed tolerance. The tolerance amount is programmable and in one embodiment, the tolerance amount is 0 and in another embodiment the tolerance amount is 1. If the second error sum is greater than the tolerance, then errors are found within the second error template **380** and at step **345** the current frame is muted (whether or not the current frame has an error therein). After muting, step **305** is entered to obtain and process the next frame. At step **340**, the second error

sum is not greater than the tolerance value and the normal decode process is performed on the current frame with an applied recovery stage. It is appreciated that step **340** is similar to step **320** except step **340** includes a recovery stage because at least one error was seen in the first template **360** and therefore a recovery from this error is being processed. After step **340**, step **305** is entered again to obtain and process the next encoded frame.

It is appreciated that if the second error summation is greater than the tolerance, then the current frame is skipped and the output is muted (whether or not the current frame contains an error therein), otherwise, the current frame is normally decoded and played. In this way, the number of transition times from normal play to mute and from mute to normal play (unmute) is reduced. In effect, the muting strategy is extended across several non-error frames depending on the accumulated error rate so that short mutings are merged into a long muting. When the error rate is high, process **280** acts to merge together adjacent error frames (mute merging) by increasing the error recovery delay period. The amount of mute merging is adaptive and is based on the error rate.

At step **345**, a number of different muting operations can be performed to mute the current frame. In the preferred embodiment, a smooth muting with zeros can be applied to decline the audio signal at a given rate according to a window function and in an alternate embodiment, a frame repeat can be performed. FIG. **6** illustrates smooth muting with zeros to reduce the "pop" sounds associated with muting. In this embodiment, an attenuation or "window" function **420** is applied to the decoded audio frame represented as signal **410** to decline its amplitude. Windowing starts at the zero-cross point. The attenuation function represents the amount of the original signal **410** allowed to exist at any given time and the remainder of the audio signal is padded (e.g., replaced) with zeros to provide a mute. Smoothing functions and muting using window functions are well known.

The selection of the length of the second template **380** is made variable and adapts based on the length table indexed by the error occurrence frequency. Under the premise of merging intermittent errors over the past frames, the length of the second template **380** should be as small as possible to minimize error recovery delay. However, two competing interests need to be satisfied. On one hand, (a), when the length of the second template **380** is large, the benefit is that intermittent errors over several frames can be merged into a longer mute, but the down side is that error recovery delay is longer. On the other hand, (b), when the length is small, the down side is that intermittent errors are not merged and this causes intermittent sound, but the benefit is that error recovery delay is shorter. To satisfy both of these interests, the following relationships can be used. To satisfy (a), the template length of template **380** plus the sum_error1 should be greater than or equal to (Y+1) where Y was the length of the fixed template **360**. To satisfy (a) and (b), the template length of template **380** should be equal to (Y+1−sum_error1). These relationships are used to determine the entries of the table length lookup table.

It is appreciated that process **280**, while described with respect to the AC3 data format, can also be applied to other audio encoding formats such as MPEG audio, AAC and DV audio, etc.

Frame Repeat for Single Frame Mutes

Error concealment can be performed in lieu of soft muting in cases where there are only 1 or 2 error frames in a row because error concealment, using frame repeating, is barely audible in these cases whereas soft muting often creates a small audible mute interval. Therefore, in those cases when the error rate of the bitstream **134** is not high, e.g., the frames in the neighborhood of the error frame have a few to no errors, a single error concealment operation is performed in accordance with one embodiment of the present invention. A single error concealment operation can be performed by repeating the previous non-error frame of the error frame. This operation can also be applied to two consecutive frame errors that follow a non-error frame. To achieve a smooth transition between the repeated frame and the previous frame, an overlap-add of the delay of the last block of the previous frame and the PCM data of the first block of the repeated frame is performed. Also, to achieve a smooth transition between the repeated frame and the following (next) frame, an overlap-add of the delay of the last block of the repeated frame and the PCM data of the first block of the following frame is performed. The implementation is performed in the time-domain rather than the code-domain as a result of certain hardware considerations. Frame repeat can be performed for two or three frames with errors therein. Applying repeating to more than three consecutive error frames can create distortion. Therefore, in these cases, the muting process as described with respect to FIG. 4 is thus applied.

FIG. **7** illustrates steps in a process **440** of one embodiment of the present invention for repeating the previous frame of an error frame to perform error concealment. The replication is performed in the time domain and special data manipulations are performed to produce smooth signal transition at the frame interfaces. Process **440** can be realized in hardware or it can be realized in software. As software, process **440** is realized as instruction code executed by system **112** (FIG. **15**). At step **445**, a next audio encoded frame of information is received from the input bitstream **134** and is referenced as the current encoded frame (e.g., frame n). A check is made at step **450** to determine if an error is present within the current encoded frame. This determination can be made by the parser process **270**. If the current encoded frame indicates that no error is present, then step **455** is entered where a normal decode of the current encoded frame is performed and the decoded audio frame is stored in the audio output buffer **250** for playout. Step **455** is analogous to step **320** of FIG. **4**. It is appreciated that the presence of an error can be determined at step **450** using the same error detection techniques described with respect to FIG. **4**.

If an error is detected in the current encoded frame, then step **460** is entered. At step **460**, the decoded version of the previous frame is obtained from the audio output buffer **250** and the PCM data from blocks **1**–**5** of the previous frame are directly copied and used in place of blocks **1**–**5** of the current encoded frame.

FIG. **8** and FIG. **9** illustrate an example. FIG. **8** illustrates a portion **134***c* of the bitstream including a current encoded frame **512** (having a detected error), a previous decoded frame **510** (frame n-1) and two next frames **514** (n+1) and **516** (n+2) which remain encoded. At step **460**, the PCM (pulse code modulation) data associated with blocks **1**–**5** of the previous frame **510**, e.g., data **510***b*–**510***f*, are copied and used as the decoded data for the current frame **512**. FIG. **9** illustrates this replacement with the PCM data **513***b*–**513***e* (of the repeated frame **512'**) being a direct copy of PCM data **510***b*–**510***e* of the previous frame **510**. The PCM data for the previous frame **510** is obtained from the audio output buffer **250** because this frame **510** has already been decoded by decoder **200**. As shown in FIG. **9**, the resultant modified

current frame 512' (now called the repeated frame) contains the same PCM data representative of blocks 1–5 as the previous frame 510.

At step 465 of FIG. 7, the delay data (from the delay array) associated with the last block (AB5) 510f of the previous frame 510 is obtained and data shuffling is performed on this delay data. The delay array is a specified data structure that for use in decoding next frames and is specified by Dolby. FIG. 9 illustrates this delay data as 511. At step 470, the PCM data associated with the first block (AB0) 510a of the pervious frame 510 is accessed. At step 475, well known weighting functions are applied to the delay data 511 and to the PCM data 510a of steps 465 and 470. At step 480, the results of the weighting functions are added together (as shown in FIG. 9) and stored as the resultant PCM data used for the first block 513a of the repeated frame 512'. Once the repeated frame 512' has been fully constructed with PCM data 513a–513f, it is forwarded to the audio output buffer 250 for playout. The result is that the delay data associated with the last block of the previous frame 510 is added and overlapped with the first block of the repeated frame 512' to smooth out the interface between these frames. Since only the decoded PCM data of the previous frame 510 is used above, the compressed code of the previous frame 510 is not necessary for process 440 and time-consuming decoding processes are not used, but rather, what is used is a weighted overlap-add function. It is appreciated that if the next frame (frame 514) is also in error, the above process 440 can be repeated for this next frame.

In an alternative embodiment, the same function can be applied to the interface between the repeated frame 512' and the next frame 514. More specifically, this embodiment of the present invention also adds the delay associated with the last block (AB5) of the repeated frame 512' with the PCM data associated with the first block (AB0) of the next frame 514 (with appropriate data shuffling and weighting) to smooth the interface between these frames.

Reduced Audio Frame Over-Run in Muting Operation

An embodiment of the present invention provides a method for reducing the number of audio frames that are played out subsequent to a mute command. A mute command can arise incident to a channel change command, e.g., a viewer decides to change a watched channel from channel A to channel B. When the change channel command is received by the decoder 200, it immediately freezes the current video frame as indicated by the read pointer of the video output buffer. The audio output, however, cannot be stopped immediately because it is synchronized as the slave to the video signal and the respective durations of the audio and video frames are different. This embodiment of the present invention reduces the number of audio frame overruns, that is, the number of audio frames that are played out subsequent to the video freeze in a channel change situation.

FIG. 12 illustrates an exemplary audio output buffer 250 containing storage (entries 250a–250g) for at least 7 decoded audio frames (descriptor0–descriptor6). Although not shown, there is a corresponding video output buffer. The just decoded audio frames are stored at the write pointer 770 and the audio frames to be played out are read from the read pointer 760 of the audio output buffer 250. The audio output buffer 250 is a circular buffer and therefore the read and write pointers are cyclic. After a read or a write, the corresponding pointer is incremented by one. There is a difference between the read and write pointers of about three to four entries in the buffer 250 to account for the well known delay or "lag" between the video and the audio information. Because it can take a relatively long time to restart the read and write pointers to their proper positions and update sequences, it is not desirable to halt the read and write pointers in response to a mute command. If this were done, there would be a slight delay noticed upon entering the next channel (e.g., channel B) after a channel change while these pointers become re-initialized. Therefore, this embodiment of the present invention provides a method for reducing audio over-run without halting the operation of the read and write pointers.

FIG. 11 illustrates a decoder unit 200 in accordance with this embodiment of the present invention. The decoder unit 200a is similar to the decoder 200 of FIG. 3B except for a channel change detect logic block 715 which generates control signals 712 to a first zero block 710 and also generates control signals 714 to a second zero block 720. The first zero block 710 is responsible for zeroing the encoded audio bitstream subsequent to a mute command received over line 712. By zeroing the input audio encoded bitstream when a mute command is received, this effectively will provide zeroed audio frames starting from the position of the write pointer 770 of the audio output buffer 250. This is shown by FIG. 13 with frames 4–7 being zeroed. FIG. 13 assumes a mute command was received when the write pointer 770 was at frame 4. Therefore, the decoder 200a gets system commands from a command module to either decode the next audio frame or mute all the preceding frames in the decoder 200a.

The second zero block 720 also receives a mute command over line 714 and functions to zero all frames between (1) the write pointer 770 and (2) two frames above the write pointer 770. In the example of FIG. 13, the frames that are zeroed by the second zero block 720 are frame 2 250c and frame 3 250d. The second zero block 720 does not zero frame 0 of FIG. 13 because the read pointer 760 may be pointing on this frame and playing it out when the channel change occurs. Frame 1 may or may not be windowed to smoothen the audio. Although frames are zeroed in the audio output buffer 250, the write and read pointers are allowed to run normally. FIG. 13 therefore illustrates an exemplary state of the audio output buffer 250 subsequent to a mute command in accordance with this embodiment of the present invention. In this example, at most two decoded audio frames will be played out subsequent to the mute command (e.g., frame 0 and frame 1). It is appreciated that two audio frames (e.g., 64 milliseconds in duration together) is not typically audible.

FIG. 10 illustrates the steps in accordance with this embodiment of the present invention. Process 600 can be realized in hardware or it can be realized in software. As software, process 600 is realized as instruction code executed by system 112 (FIG. 15). Step 610 looped until a channel change is detected or otherwise an audio mute is required. At step 615, the current video frame being output by the video output buffer 250 is held thereby freezing the frame on the display device or monitor. At step 620, an audio bitstream of the decoder is zeroed, thereby causing the decoded audio data (associated with the old channel) that is present within the audio buffer 250 to become zeroed starting at the write pointer position. As discussed above, this will zero all frames of the audio output buffer 250 starting from the write pointer 770 and counting down the buffer. At step 625, the decoder directly zeros two decoded audio frames above the write buffer. A number of well

known windowing functions can be applied to these frames to perform the zeroing operation. As discussed above, with respect to FIG. 13, step 625 effectively zeros frames 2 and 3. In the typical case, there are four audio frames between the read and write pointer 770, so frames 2 and 3 also represent the second and third frames away from the read pointer 760. The audio decode and playback processes are then allowed to operate normally and process 600 returns to step 610.

The above process 600 can be applied to a number of well known encoding standards, for example, AC3, AAC, MPEG-Audio and DV audio. Process 600 can also be applicable in a situation where the audio interface is supposed to be transmitting zero data right from the system boot-up even when there is no actual data (e.g., IEC60958). A variation of the above approach can be used to implement that condition. Process 600 can be used by any audio interfaces transmitting linear PCM data, for instance, ACLINK, IEC60958, or IIS.

FIG. 14 illustrates a timing diagram of the above operations. Signal 816 is the channel change command which simultaneously generates an audio mute signal at the indicated pulse. Signal 818 represents the state of the decoding logic and, as shown, it drops down three cycles after the mute command pulse of signal 816. This represents the input audio encoded bitstream being zeroed by the first zero logic 710. Signal 820 represents the video display and subsequent to the mute command pulse of signal 816, it enters a freeze frame as shown by interval 830. Signal 822 illustrates the operation of the audio playback of the prior art method and includes four audio over-run frames 835 that are played out after the video freeze commences. Signal 824 illustrates the operation of the audio in accordance with this embodiment of the present invention. In accordance with signal 824, only two audio frames 840 are played out subsequent to the start of the video freeze. By reducing the audio frame over-run by at least two frames, the present invention is able to eliminate the annoying and confusing sounds that often result from a channel change operation of the prior art.

Computer System Platform

Embodiments of the present invention can be implemented within a computer system. FIG. 15 illustrates a computer system 112 that can be a general purpose computer system or it can be an embedded system within an electronic device, such as an intelligent device, an AV decoder system, a set-top-box, a receiver unit, a digital television unit, etc. Computer system 112 includes an address/data bus 100 for communicating information, a central processor 101 coupled with the bus for processing information and instructions, a volatile memory 102 (e.g., random access memory RAM) coupled with the bus 100 for storing information and instructions for the central processor 101 and a non-volatile memory 103 (e.g., read only memory ROM) coupled with the bus 100 for storing static information and instructions for the processor 101. Computer system 112 also includes a data storage device 104 ("disk subsystem") such as a magnetic or optical disk and disk drive coupled with the bus 100 for storing information and instructions and a display device 105 coupled to the bus 100 for displaying information to the computer user.

Also included in computer system 112 of FIG. 15 is an optional alphanumeric input device 106 including alphanumeric and function keys coupled to the bus 100 for communicating information and command selections to the central processor 101. System 112 also includes an optional cursor control or directing device 107 coupled to the bus for communicating user input information and command selections to the central processor 101. The cursor directing device 107 can be implemented using a number of well known devices such as a mouse, a track ball, a track pad, an electronic pad and stylus, an optical tracking device, a touch screen etc. The display device 105 utilized with the computer system 112 is optional and may be a liquid crystal device, cathode ray tube (CRT), field emission device (FED, also called flat panel CRT) or other display device suitable for creating graphic images and alphanumeric characters recognizable to the user.

The preferred embodiment of the present invention, a digital audio decoder system for a multimedia information system having improved error concealment functionality, improved muting capabilities and reduced audio frame overrun in response to a mute command, is thus described. While the present invention has been described in particular embodiments, it should be appreciated that the present invention should not be construed as limited by such embodiments, but rather construed according to the below claims.

What is claimed is:

1. A method for muting a portion of an encoded bitstream of audio information comprising the steps of:

a) with respect to a current encoded audio frame of said encoded bitstream, computing a length of a dynamic template based on an error rate of said encoded bitstream, said dynamic template encompassing a plurality of previous encoded frames of said encoded bitstream;

b) summing errors of said plurality of previous encoded frames within said dynamic template to produce a first error sum;

c) determining if said first error sum exceeds a prescribed tolerance; and

d) adaptively merging muted error frames by muting said current encoded audio frame provided said first error sum value exceeds said prescribed tolerance whether or not said current encoded audio frame has an error.

2. A method as described in claim 1 wherein said step a) comprises the steps of:

a1) with respect to said current encoded audio frame, summing errors of a plurality of previous encoded frames encompassed by a fixed-length template to produce a second error sum; and

a2) using said second error sum as an index to a look-up table to compute said length of said dynamic template.

3. A method as described in claim 2 wherein said step a1) and said step b) are performed using an error array which contains a respective bit for each encoded audio frame indicating whether or not an error resides within its associated encoded audio frame.

4. A method as described in claim 2 wherein said plurality of previous encoded frames encompassed by said fixed-length template are measured from and include said current encoded audio frame and wherein said plurality of previous encoded frames encompassed by said dynamic template are measured from and include said current encoded audio frame.

5. A method as described in claim 4 wherein said fixed-length template is 24 audio frames in length and said tolerance is 1.

6. A method as described in claim 2 wherein steps a2), b), c) and d) are bypassed if said second error sum is zero.

7. A method as described in claim 2 wherein said encoded bitstream of audio information is substantially compliant with the AC3 digital audio standard.

8. A method for muting a portion of an encoded bitstream of audio information comprising the steps of:

a) detecting if a current encoded audio frame of said encoded bitstream contains an error; and

b) provided an error is detected, repeating a previous decoded audio frame in lieu of said current encoded audio frame, said repeating comprising the steps of:

b1) obtaining decoded data of said previous audio frame;

b2) generating a repeated audio frame by replicating said decoded data of said previous audio frame for use in lieu of said current encoded audio frame;

b3) modifying said repeated audio frame by adding delay information of a last block of said previous audio frame with pulse code modulated (PCM) data of a first block of said repeated audio frame to generate new decoded data for said first block of said repeated audio frame; and

b4) sending said repeated audio frame to an audio output buffer for playout.

9. A method as described in claim 8 wherein said step b1) obtains said decoded data from said audio output buffer.

10. A method as described in claim 9 wherein said step b3) comprises the steps of:

shuffling and weighting said delay information to generate shuffled and weighted delay information;

weighting said PCM data to generate weighted PCM data;

adding said shuffled and weighted delay information with said weighted PCM data to generate said new decoded data for said first block of said repeated audio frame.

11. A method as described in claim 9 further comprising the steps of:

c) provided an error is detected in a next encoded audio frame immediately following said current encoded audio frame, repeating said current encoded audio frame in lieu of said next encoded audio frame; said step c) comprising the steps of:

c1) obtaining decoded data of said current encoded audio frame;

c2) generating a second repeated audio frame by replicating said decoded data of said current encoded audio frame for use in lieu of said next encoded audio frame;

c3) modifying said second repeated audio frame by adding delay information of a last block of said current encoded audio frame with pulse code modulated (PCM) data of a first block of said second repeated audio frame to generate new decoded data for said first block of said second repeated audio frame; and

c4) sending said second repeated audio frame to an audio output buffer for playout.

12. A method as described in claim 8 further comprising the steps of:

c) provided an error rate of said encoded bitstream is high, performing mute merging, said step c) comprising the steps of:

c1) with respect to said current encoded audio frame of said encoded bitstream, computing a length of a dynamic template based on an error rate of said encoded bitstream, said dynamic template encompassing a plurality of previous encoded frames of said encoded bitstream;

c2) summing errors of said plurality of previous encoded frames within said dynamic template to produce a first error sum;

c3) determining if said first error sum exceeds a prescribed tolerance; and

c4) adaptively merging muted error frames by muting said current encoded audio frame provided said first error sum value exceeds said prescribed tolerance whether or not said current encoded audio frame has an error.

13. A method as described in claim 12 wherein said step c1) comprises the steps of:

with respect to said current encoded audio frame, summing errors of a plurality of previous encoded frames encompassed by a fixed-length template to produce a second error sum; and

using said second error sum as an index to a look-up table to compute said length of said dynamic template.

14. A method as described in claim 8 wherein said encoded bitstream of audio information is substantially compliant with the AC3 digital audio standard.

* * * * *

US005684829A

United States Patent [19]

[11] Patent Number: 5,684,829

[45] Date of Patent: Nov. 4, 1997

Kizuki et al.

[54] DIGITAL SIGNAL PROCESSING CODING AND DECODING SYSTEM

[75] Inventors: Takafumi Kizuki, Yokosuka; Toshihiro Maruyama, Kawasaki; Susumu Takahashi, Tokyo, all of Japan

[73] Assignee: Victor Company of Japan, Ltd., Tokyo, Japan

[21] Appl. No.: 582,696

[22] Filed: Jan. 4, 1996

[30] Foreign Application Priority Data

Jan. 27, 1995 [JP] Japan 7-31435

[51] Int. Cl.^{6} H04B 14/04

[52] U.S. Cl. 375/242; 341/64

[58] Field of Search 375/242, 243, 375/244, 245, 254, 246–253, 296, 340, 316; 341/50, 64, 143, 200, 51, 52; 370/10, 69.1, 74, 77, 98, 110.1, 110.4

[56] References Cited

U.S. PATENT DOCUMENTS

5,444,741 8/1995 Mahieux 375/254

5,550,812 8/1996 Phillips 375/362

OTHER PUBLICATIONS

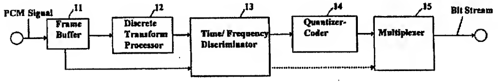
On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform—Proceedings of the IEEE, vol. 66, No. 1 Jan. 1978, Fredric J. Harris, Member, IEEE.

Primary Examiner—Stephen Chin
Assistant Examiner—Hai H. Phan
Attorney, Agent, or Firm—Michael N. Meller

[57] ABSTRACT

A signal coding system capable of high efficiency, high quality signal coding is provided. Digital signals represented in the time domain are divided into set time interval data units and output. One output is converted to a digital signal represented in the frequency domain, and the other is output as-is. The energy dispersion of the digital signal represented in the frequency domain is compared with that of the digital signal represented in the time domain, and the digital signal having the least energy dispersion is coded. This coded digital signal is then multiplexed with an identification signal to identify it as a frequency domain or time domain signal, and the resulting multiplexed signal is output.

7 Claims, 8 Drawing Sheets



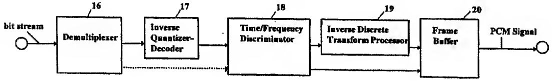
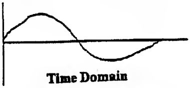


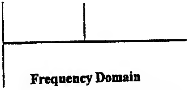
FIG. 1(A)
PRIOR ART

Constant-type Sound
(e.g. a wind and string instrument)



Time Domain
Energy Dispersion: Large

FIG. 1(B)
PRIOR ART



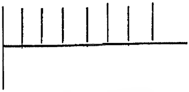
Frequency Domain
Energy Dispersion: Small

FIG. 4(A)
PRIOR ART

Impulse-type Sound (e.g. Percussion instrument)

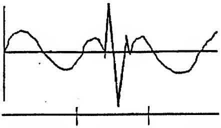
Time Domain
Energy Dispersion: Small

FIG. 4(B)
PRIOR ART



Frequency Domain
Energy Dispersion: Large

FIG. 8



First Frame Second Frame Third Frame
Frequency Domain Frequency Domain
Time Domain

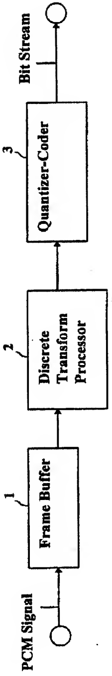

PCM Signal
1
Frame Buffer
2
Discrete Transform Processor
3
Quantizer-Coder
Bit Stream

FIG. 2

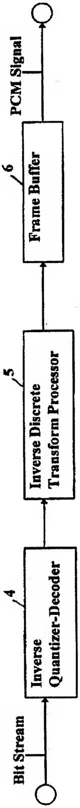

Bit Stream
4
Inverse Quantizer-Decoder
5
Inverse Discrete Transform Processor
6
Frame Buffer
PCM Signal

FIG. 3

FIG. 5

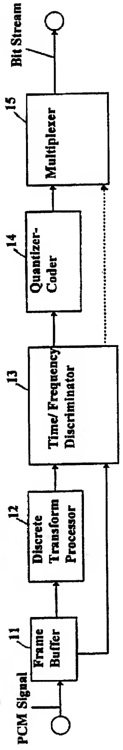
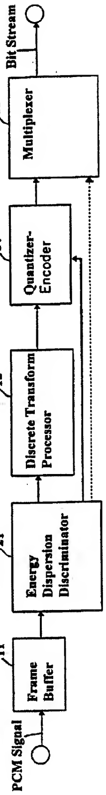

PCM Signal
11
Frame Buffer
12
Discrete Transform Processor
13
Time/ Frequency Discriminator
14
Quantizer-Coder
15
Multiplexer
Bit Stream

FIG. 6


PCM Signal
11
Frame Buffer
21
Energy Dispersion Discriminator
12
Discrete Transform Processor
14
Quantizer-Encoder
15
Multiplexer
Bit Stream

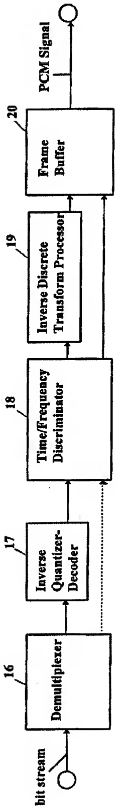

bit stream
16
Demultiplexer
17
Inverse Quantizer-Decoder
18
Time/Frequency Discriminator
19
Inverse Discrete Transform Processor
20
Frame Buffer
PCM Signal

FIG. 7

FIG. 9

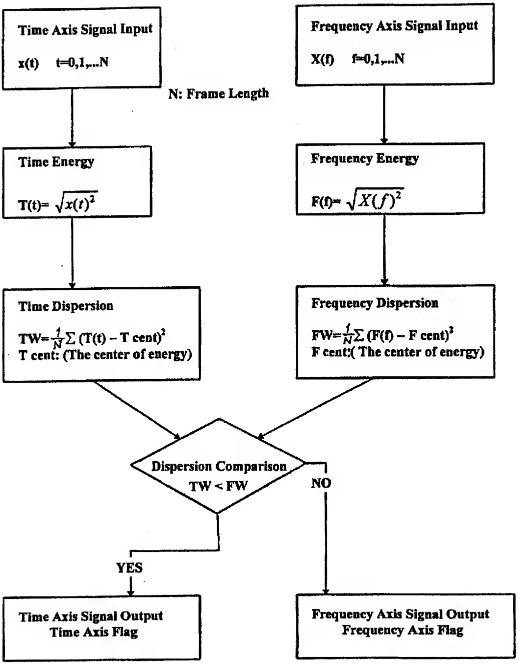

Time Axis Signal Input
x(t) t=0,1,...N
Frequency Axis Signal Input
X(f) f=0,1,...N
N: Frame Length
Time Energy
$T(t)=\sqrt{x(t)^2}$
Frequency Energy
$F(f)=\sqrt{X(f)^2}$
Time Dispersion
$TW=\frac{1}{N}\Sigma(T(t)-T\ cent)^2$
T cent: (The center of energy)
Frequency Dispersion
$FW=\frac{1}{N}\Sigma(F(f)-F\ cent)^2$
F cent:(The center of energy)
Dispersion Comparison
TW < FW
NO
YES
Time Axis Signal Output
Time Axis Flag
Frequency Axis Signal Output
Frequency Axis Flag

FIG. 10

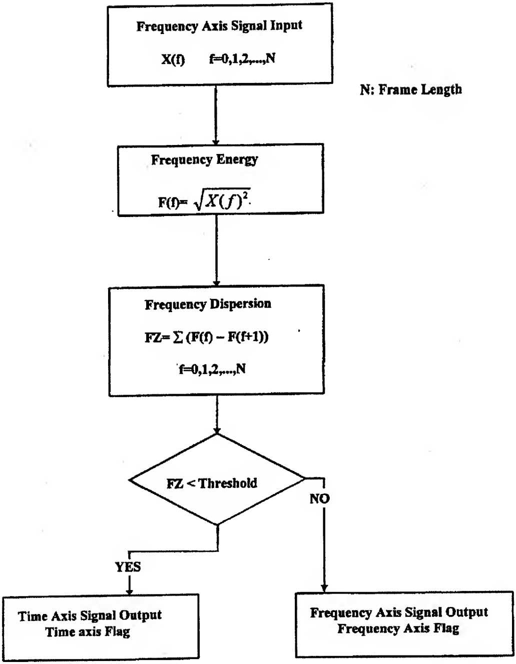

Frequency Axis Signal Input
X(f) f=0,1,2,...,N
N: Frame Length
Frequency Energy
F(f)= $\sqrt{X(f)^2}$
Frequency Dispersion
FZ= Σ (F(f) – F(f+1))
f=0,1,2,...,N
FZ < Threshold
NO
YES
Time Axis Signal Output
Time axis Flag
Frequency Axis Signal Output
Frequency Axis Flag

FIG. 11

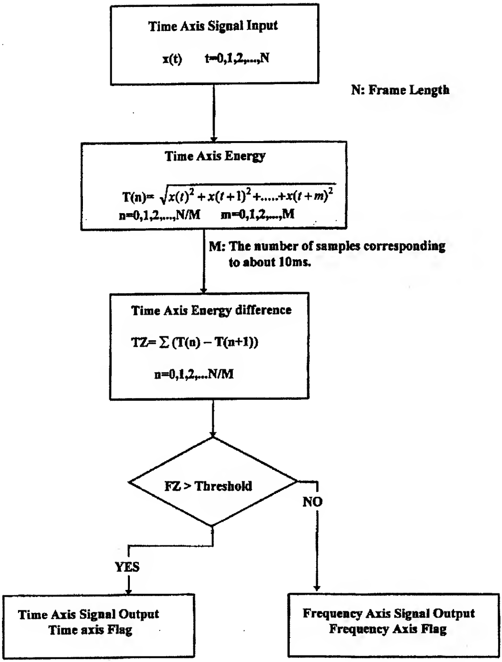
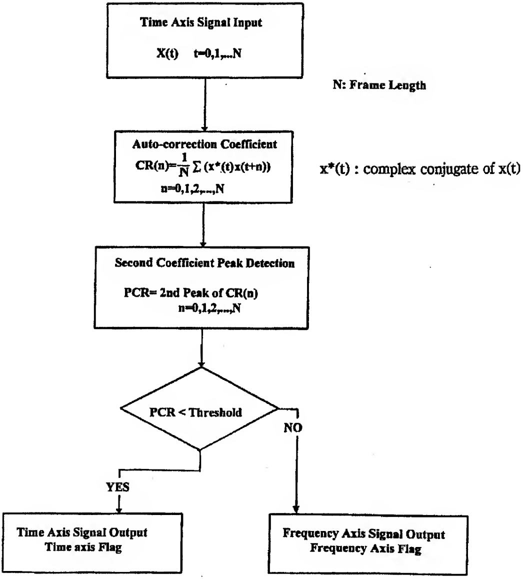

Time Axis Signal Input
x(t) t=0,1,2,...,N
N: Frame Length
Time Axis Energy
T(n)= $\sqrt{x(t)^2+x(t+1)^2+.....+x(t+m)^2}$
n=0,1,2,...,N/M m=0,1,2,...,M
M: The number of samples corresponding to about 10ms.
Time Axis Energy difference
TZ= Σ (T(n) − T(n+1))
n=0,1,2,...N/M
FZ > Threshold
NO
YES
Time Axis Signal Output
Time axis Flag
Frequency Axis Signal Output
Frequency Axis Flag

FIG. 12


Time Axis Signal Input
X(t) t=0,1,...N
N: Frame Length
Auto-correction Coefficient
CR(n)=$\frac{1}{N}\Sigma$ (x*(t)x(t+n))
n=0,1,2,...,N
x*(t) : complex conjugate of x(t)
Second Coefficient Peak Detection
PCR= 2nd Peak of CR(n)
n=0,1,2,...,N
PCR < Threshold
NO
YES
Time Axis Signal Output
Time axis Flag
Frequency Axis Signal Output
Frequency Axis Flag

DIGITAL SIGNAL PROCESSING CODING AND DECODING SYSTEM

BACKGROUND OF THE INVENTION

1. Field of the Invention

This invention relates to signal encoding and decoding system that compresses and decompresses the information content of a pulse-code-modulated (PCM) digital signal.

2. Description of the Prior Art

There exists in the art, as a method of converting an analog audio signal into a digital signal, a time domain representation method wherein the amplitude of the analog signal is quantized into discrete units of quantity by sampling it at fixed time intervals.

With such methods that represent signals in the time domain only, however, the volume of data in the resulting digital signal is large, and neither the data transmission signal band nor the storage capacity of the storage media can be reduced. There are, therefore, a number of methods that may be considered for compressing such digital signals in the time domain.

The main methods for audio signal coding are subband coding (SBC), in which the signal is coded by dividing the signal into subbands, and advanced transformation coding (ATC) in which the signal is coded by adaptive transformation. In both SBC and ATC coding, an audio signal input as a time series (time domain) signal is transformed to the frequency domain, and then coded using the uneven dispersion of energy across a wide band in the frequency domain.

That is, as shown in FIG. 1(A), the energy of a constant-type sound signal, such as that produced by a wind or string instrument, is widely dispersed in the time domain, and a large amount of data would therefore be required to code it, If, however, a discrete transform of the above constant-type tone is taken to convert it to the frequency domain, as shown in FIG. 1(B), the extent of the energy dispersion is small, and it therefore requires only a small amount of data to express it.

In this coding method, data compression is performed by allocating a large number of bits (information) to the coding of frequency bands (subbands) that have a large amount of energy, and few bits to subbands that have little energy, or that are audibly unimportant.

FIG. 2 shows an example of a signal coding system that performs such transform coding, and FIG. 3 shows an example of a system that decodes the resulting signal. These systems are described below.

In the system of FIG. 2, an input digital audio signal (PCM signal) represented in the time domain is supplied to frame buffer 1, where it is windowed weighted by a window function and output, frame-by-frame.

In the windowing process, window functions such as Hanning Window and Hamming Window are applied to the input audio signal of a continuous time-series signal to weight its amplitude, and is divided into "frames," the units in which subsequent signal processing is performed (see F. J. Harris, "On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform.", Proc. IEEE, vol.66, no. 1, pp.51–83, 1978; Mikio Takagi and Haruhisa Shimoda, "Gazoh Kaiseki Handbook", Tokyo Daigaku Shuppan, pp20–25, 1991).

The output of frame buffer 1 is then supplied frame-by-frame to discrete transform processor 2, where a discrete transform such as the discrete cosine transform (DCT), discrete Fourier transform (DFT), Karhunen-Loeve transform (KLT) etc. is performed on the signal, to transform it to the frequency domain. Finally, in quantizer-coder 3, the output of discrete transform processor 2 is converted to a bit stream in which most of the bits are allocated to portions of the frequency spectrum that contain large amounts of energy, or that are audibly important.

With transform coding, when the signal is transformed to the frequency domain, the more uneven the dispersion of energy across the spectrum, the higher the compression ratio. It is therefore desirable to use the discrete transform with the highest transformation efficiency. The KLT transform has the highest "ideal" efficiency, but in terms of practical efficiency (number of calculations, etc.) it is about the same as the DCT. Therefore, the transform that is normally used is actually the DCT, which has the highest to computation speed.

In the signal decoding system shown in FIG. 3, the bit stream received at the decoding system input is a digital audio signal represented in the frequency domain. This input is supplied to inverse quantizer-decoder 4, where it is decoded. The output of inverse quantizer-decoder 4 is fed to inverse discrete transform processor 5, where its inverse discrete transform is returned to the time domain; i.e. the inverse discrete cosine transform (IDCT), inverse discrete Fourier transform (IDFT), or inverse Karhunen-Loeve transform (IKLT), etc., as applicable, is transformed. The output of inverse discrete transform processor 5 is inverse-windowed by frame buffer 6, and output as a decoded digital audio signal represented in the time domain.

The inverse windowing process multiplies each frame of the signal by the inverse of the function used to window it, thereby restoring the amplitude of the audio signal to its original state removing the window components.

Thus the information content of a constant-type tone audio signal, as shown in FIG. 1(A) and FIG. 1(B), can be compressed by performing a discrete transform to translate the signal to the frequency domain. As shown in FIG. 4(A), however, in impulse-type sound signals such as produced by percussion instruments, the energy dispersal in the time domain is small and the energy is unevenly distributed. If the discrete transform of this type of audio signal is taken, to translate it to the frequency domain, the energy will be widely dispersed, as shown in FIG. 4(B). This was a problem with the conventional system, in that for this type of signal, rather than being improved, the compression efficiency was actually reduced.

Another problem with this system was that in impulse-type sound signals, when portions having abrupt energy changes were coded in the freqnency domain, a type of noise referred to as "pre-echo noise" was produced in the low energy portions of the signal, degrading the coding quality.

BRIEF SUMMARY OF THE INVENTION

1. Object of the Invention

It is the object of this invention to effect high efficiency, high quality signal coding and decoding by switching between the time and frequency domains of the digital signal, depending on the nature of the input digital signal, to perform the coding.

2. Brief Summary

Provided, according to a first aspect of this invention, is a signal coding system for coding an input digital signal, comprising: a data accumulation means for dividing an input digital signal represented in a time domain into set time

intervals, and outputting said signal; a discrete transform processing means for transforming a digital signal received from said data accumulation means into a digital signal represented in a frequency domain; a discrimination means for determining whether an input digital signal is a constant-type digital signal or an impulse-type digital signal, and for concurrently outputting an identification signal indicating the result of this determination; a coding means that if, based on the identification signal supplied from said discrimination means, said input digital signal is found to be a constant-type signal, codes said digital signal converted by said discrete processing means for representation in said frequency domain, and if said input digital signal is found to be an impulse-type sound-type signal, codes said digital signal represented in said time domain; and a multiplexing means for multiplexing said digital signal coded by said coding means with said identification signal.

Further provided, according to a second aspect of this invention, is a signal coding system for coding input digital signals comprising: a data accumulation means for dividing an input digital signal represented in a time domain into set time intervals, and outputting said signal; a discrimination means for determining whether a digital signal received from said data accumulation means is a constant-type digital signal or an impulse-type digital signal, and for concurrently outputting an identification signal indicating the result of this determination; a discrete transform processing means for transforming a digital signal received from said discrimination means into a digital signal represented in a frequency domain; a coding means that if, based on the identification signal received from said discrimination means, said input digital signal is found to be a constant-type signal, codes said digital signal converted by said discrete processing means for representation in said frequency domain, and if said input digital signal is found to be an impulse-type sound-type signal, codes said digital signal represented in said time domain; and a multiplexing means for multiplexing said digital signal coded by said coding means with said identification signal.

Still further provided, according to a third aspect of this invention, is a signal decoding system for decoding a digital signal divided into set time intervals containing a mixture of digital signals represented in the frequency domain and digital signals represented in the time domain and coded in this mixed state, and also having multiplexed therein, identification signals that identify the content of each time interval as either a time domain or a frequency domain signal, comprising: a separation means for separating an input digital signal into said coded digital signal and said identification signal portions; a decoding means for decoding said coded digital signal received from said separation means; a discrimination means for determining whether a digital signal received from said decoding means is represented in the frequency domain or in the time domain, based on said identification signal received from said separation means; an inverse discrete transform processing means for converting a digital signal received from said discrimination means represented in the frequency domain to a digital signal represented in the time domain; and an output means for outputting, in time series sequence, digital signals represented in the time domain received from said inverse discrete transform processing means, and digital signals represented in the time domain received from said discrimination means.

The invention determines, for each transform coding frame, whether the sound represented therein is constant-type sound or impulse-type sound by comparing the extent of its energy dispersion in the frequency domain with the extent of its energy dispersion in the time domain, and codes the data in the frequency domain if it is constant-type sound, and codes it in the time domain if it is impulse-type sound, and by so doing, improves the coding quality and coding efficiency over that which could be realized by coding in the frequency domain only.

The above and other related objects and features of the invention will be apparent from a reading of the following description of the found in the accompanying drawings, and the novelty thereof pointed out in the appended claims.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1(A) is a waveform diagram showing the waveform of a constant-type sound signal, and in particular, the waveform of the signal in the time domain.

FIG. 1(B) is a waveform diagram showing the waveform of a constant-type sound signal, and in particular, the waveform of the signal in the frequency domain.

FIG. 2 is a block diagram showing an example of a signal coding system that performs transform coding.

FIG. 3 is a block diagram showing an example of a signal decoding system.

FIG. 4(A) is a waveform diagram showing the waveform of an impulse-type sound signal, and in particular, the waveform of the signal in the time domain.

FIG. 4(B) is a waveform diagram showing the waveform of an impulse-type sound signal, and in particular, the waveform of the signal in the frequency domain.

FIG. 5 is a block diagram showing one embodiment of the signal coding system of the present invention.

FIG. 6 is a block diagram showing another embodiment of the signal coding system of the present invention.

FIG. 7 is a block diagram showing one embodiment of the signal decoding system of the present invention.

FIG. 8 is a waveform diagram showing an example of a signal having a mixture of constant-type sound and impulse-type sound.

FIG. 9 is a diagram for explaining one example of a time frequency discriminator as shown in FIG. 5.

FIG. 10 is a diagram for explaining another example of a time frequency discriminator as shown in FIG. 5.

FIG. 11 is a diagram for explaining one example of an energy dispersion detector as shown in FIG. 6.

FIG. 12 is a diagram for explaining another example of an energy dispersion detector as shown in FIG. 6.

DETAILED DESCRIPTION OF THE INVENTION

The preferred embodiment of the present invention is described in detail below, based on the accompanying drawings.

One Embodiment of the Signal Coding System

FIG. 5 is a block diagram showing one embodiment of the signal coding system of the present invention. In FIG. 5, the input digital audio signal represented in the time domain is supplied to frame buffer 11, where it is windowed frame-by-flame, and output.

In the windowing process, the input audio signal (a continuous time-series signal) is multiplied by window functions such as Harming window and Hamming window to weight its amplitude, and is then divided into "frames,"

which are the data units on which subsequent signal processing will be performed(see F. J. Harris, "On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform.", Proc. IEEE, vol.66, no. 1, pp.51–83, 1978; Mikio Takagi and Haruhisa Shimoda, "Gazoh Kaiseki Handbook", Tokyo Daigaku Shuppan, pp.20–25, 1991).

One of the outputs of frame buffer **11** is supplied frame-by-frame to discrete transform processor **12**, where a discrete transform such as the discrete cosine transform (DCT), discrete Fourier transform (DFT), Karhunen-Loeve transform (KLT) etc. is performed on the signal to map it to the frequency domain, after which it is output to time/frequency discriminator **13**. The other output of frame buffer **11** is sent, still a time domain signal, to time/frequency discriminator **13**.

Time/frequency discriminator **13** compares the energy dispersion in the frequency domain signal received from discrete transform processor **12** with the energy dispersion in the time domain signal received directly from frame buffer **11**, and outputs, to quantizer-coder **14**, the signal having less widely dispersed energy. At the same time, time/frequency discriminator **13** also outputs an identification flag to multiplexer **15** to identify the signal being sent to as a time domain signal or a frequency domain signal

Now, the two signals input to time/frequency discriminator **13**, are, as indicated in FIG. **9**, a time axis signal x(t) {t=0, 1, ... , N (where N is the frame length)}, and a frequency axis signal X(f) {f=0, 1, . . . , N (where N is the frame length)}, each having time energy T(t), and frequency energy S(f), respectively, which are given by the following equations:

$$T(t)=\sqrt{x(t)^2} \tag{1}$$

$$F(f)=\sqrt{X(f)^2} \tag{2}$$

Also, time dispersion TW and frequency dispersion FW are given by the following equations:

$$TW=\frac{1}{N}\,\Sigma\,(T(t)-T\,cent)^2 \tag{3}$$

where Tcent is the center of energy concentration in the time domain

$$FW=\frac{1}{N}\,\Sigma\,(F(f)-Fcent)^2 \tag{4}$$

where Fcent is the center of energy concentration in the frequency domain

The above "time dispersion" value indicates the time variance of the energy content of the frame with respect to the center of energy concentration on the time axis, and the "frequency dispersion" value indicates the frequency variance of the energy content of the frame with respect to the center of energy concentration on the frequency axis.

The magnitudes of the time dispersion TW and frequency dispersion FW determined as described above are then compared, and the signal in the domain having less energy dispersion is output, along with its corresponding flag. That is, if the time dispersion (TW) is less, the time axis signal and flag are output, and if the frequency dispersion (FW) is less, the frequency axis signal and flag are output.

Since time/frequency discriminator **13** of FIG. **5** outputs the signal in the domain having the least energy dispersion (frequency or time), then a) if the input is a constant-type sound signal, the frequency domain signal will be selected, and b) if the input is an impulse-type sound signal, the time domain signal will be selected.

Accordingly, if the system received an audio input signal containing a mix of both constant-type and impulse-type sound-type signals, such as that shown in FIG. **8**, the output signal selected for the first and third frames would be the frequency domain signal, and that selected for the second frame output would be the time domain signal.

In quantizer-coder **14**, quantization is performed such that portions of the input signal spectrum that have a large mount of energy and portions that are important for auditory perception are allocated most of the available bits, and the resulting signal is then output to multiplexer **15**.

Multiplexer **15** multiplexes the time/frequency identification flag received from time/frequency discriminator **13** frame-by-frame with the Signal received from quantizer-coder **14**, and outputs the result as a bit stream. The time/frequency identification flag consists of one bit header, which added ahead of the data bits.

As described above, then, the signal coding system of this invention is capable of performing efficient coding of audio signals containing a mixture of constant-type and impulse-type sound components. Also, since impulse-type sound, which gives rise to abrupt energy changes, is coded in the time domain, the disturbances referred to as pre-echo noise do not occur, thus preventing the degradation of quality normally associated therewith.

Another Embodiment of the Signal Coding System

Next, another embodiment of the signal coding system of the present invention will be explained, with reference to FIG. **6**. The parts of the system that are the same as in the embodiment that was described above using FIG. **5** are assigned the same reference numbers as in FIG. **5**, and are not discussed here.

In FIG. **6**, an input digital audio signal represented in the time domain is supplied to frame buffer **11**, where it is windowed frame-by-frame, and output. The output of frame buffer **11** is supplied to energy dispersion detector **21**.

Energy dispersion detector **21** determines whether the level of energy dispersion in the input digital audio signal is above or below a predetermined energy dispersion value (threshold level), and concurrently outputs, to multiplexer **15**, a flag indicating which of these two conditions exists.

If the energy dispersion exceeds the threshold level, the signal is determined to represent constant-type sound, in which case the output of energy dispersion detector **21** is supplied frame-by-frame to discrete transform processor **12**, where a discrete transform (DCT, DFT, KIT, etc.) of the signal is performed to map it to the frequency domain for output to quantizer-encoder **14**. If the energy dispersion is less than the threshold level, the signal represents impulse-type sound, in which case the output of energy dispersion detector **21** is sent as-is (in the time domain), to quantizer-encoder **14**.

The signal input to quantizer-coder **14** is quantized and output to multiplexer **15**. Multiplexer **15** multiplexes the time/frequency identification flag output from energy dispersion detector **21** frame-by-frame with the signal received from quantizer-coder **14**, and outputs the result as a bit stream.

Thus as explained above, the result obtained in the embodiment of FIG. **6** is the same as in the embodiment of FIG. **5**.

Other Discrimination Methods

In both of the embodiments described above, the determination as to whether the input signal represented constant

or impulse-type sound was made by detecting the amount of energy dispersion in the digital signal. This determination, however, may just as well have been performed by other methods.

In constant-type audio, for example, the decay curve of the envelope is usually gradual, and the envelope of an impulse-type signal has a sharp rising edge.

Alternate Method 1

Accordingly, the differences in the mounts of energy at various frequencies in a digital signal represented in the frequency domain can be determined. A signal with large energy differences can then be classified as a constant-type sound signal, and one in which the differences are not too great as an impulse-type signal. This method can be implemented by simply changing the time/frequency discriminator **13** in the signal coding system of FIG. **5**.

In this case, as shown in FIG. **10**, a frequency axis signal X(f) {f=0, 1, . . . , N (where N is the frame length)} is input to time/frequency discriminator **13**, and the frequency energy F(f) computed using equation (1), above. The total of the energy differences between adjacent frequency components on the frequency axis FZ is then calculated, using the following equation:

$$FZ=\Sigma(F(f)-F(F+1)) \quad (5)$$

$$f=1, 2, \ldots, N$$

The total of the energy differences calculated as indicated above is then compared with a threshold level that has been set in advance. If the total of the energy differences FZ is less than the threshold level, the signal is considered impulse-type sound, and it is output as a time axis signal, along with the corresponding flag. Conversely, if the total of the energy differences FZ exceeds the threshold level, the signal is judged as a constant-type sound signal, and is output as a frequency axis signal, along with that flag.

Alternate Method 2

In a digital signal represented in the time domain, the difference between present and preceding amplitudes can be detected, and the difference compared against a set value. Signals in which the difference falls below, and those in which the difference falls above the threshold level would then be processed as constant, and impulse-type sound signals, respectively. The signal coding system for this method can be configured by simply changing the energy dispersion detector **21**, as shown in FIG. **6**.

In this case, as indicated in FIG. **11**, in energy dispersion detector **21**, a time axis signal x(t) {t=0, 1, . . . , N (where N is the frame length)} is input, and time axis energy T(n) calculated by the following equation:

$$T(n)=\sqrt{x(t)^2+x(t+1)^2+\ldots+x(t+m)^2} \quad (6)$$

where M is the number of samples corresponding to about 10 ms.

The total of the differences between the average energy levels of adjacent fixed interval samples M (about 10 ms) on the time axis, TZ, is then calculated by the following equation:

$$TZ=\Sigma(T(n)-T(n+1)) \quad (7)$$

$$n=1, 2, \ldots N/M$$

The total of the average energy differences TZ, determined as indicated above, is then compared with a threshold level set in advance. If the total of the average energy differences TZ is less than the threshold level, the signal is considered a constant-type sound signal, and is output as a frequency axis signal, along with the corresponding flag. Conversely, if the total of the energy differences TZ exceeds the threshold level, the signal is judged an impulse-type sound signal, and is output as a time axis signal, with that flag.

Alternate Method 3

Another possible method finds the auto-correlation coefficients of the frames of a digital signal represented in the time domain. Those signals with high auto-correlation are then classified as constant-type sound, and those low auto-correlation as impulse-type sound. With this method as well, it is necessary only to change energy dispersion detector **21** of FIG. **6** to configure the signal coder system.

In this case, as indicated in FIG. **12**, in energy dispersion detector **21**, a time axis signal x(t) {t=0, 1, . . . , N (where N is the frame length, in bits)} is input, and its auto-correlation coefficient CR(n) is calculated, using the following equation:

$$CR(n)=\frac{1}{N}\Sigma(x^*(t)x(t+n)) \quad (8)$$

$$n=0, 1, 2, \ldots, N$$

x*(t): complex conjugate of x(t)

The magnitude of the coefficient's second peak PCR is then detected.

$$PCR=\text{2nd peak of CR}(n) \quad (9)$$

$$n=1, 2, \ldots, N$$

The magnitude of the detected second peak PCR is then compared with a threshold level set in advance. If the magnitude of the second coefficient peak is less than the threshold level, the signal is considered an impulse-type sound signal, and is output, along with the corresponding flag, as a time axis signal. Conversely, if the magnitude of the second peak of the coefficient PCR is greater than the threshold level, the signal is determined to be a constant-type sound signal, and is output as a frequency axis signal, along with that flag.

An Embodiment of the Signal Decoding System

An embodiment of the signal decoding system of the present invention is shown in FIG. **7** and explained below.

This signal decoding system is capable of decoding signals coded by any of the above described coding systems.

The signal decoding system of FIG. **7** is a decoding system for decoding a coded digital audio signal input received as a bit stream.

In FIG. **7**, the input signal is supplied to demultiplexer **16**, which divides the signal into the data signal and the time/frequency identification flag, which are then fed to inverse quantizer-decoder **17** and time/frequency discriminator **18**, respectively.

Inverse quantizer-decoder **17** decodes the data signal and outputs result to time/frequency discriminator **18**.

Time/frequency discriminator 18 decides whether the data signal it is receiving from inverse quantizer-decoder 17 is a frequency domain signal or a time domain signal, based on the time/frequency identification flag it receives from demultiplexer 16. If it is a frequency domain signal, it outputs it to inverse discrete transform processor 19, where an inverse discrete transform such as the inverse discrete cosine transform (IDCT), inverse discrete Fourier transform (IDFT), or the inverse Karhunen-Loeve transform (IKLT), is performed on it, to transform it to the time domain, after which it is output to frame buffer 6. If time/frequency discriminator 18 determines that the signal is a time domain signal, it outputs it as-is, directly to frame buffer 20.

Finally, the signal is inverse-windowed in frame buffer 20, and output as a digital audio signal represented in the time domain.

The inverse windowing process multiplies each frame of the signal by the inverse of the function used to window it, thereby restoring the amplitude of the audio signal to its original prewindowing state.

In this manner, the signal decoding system of the present invention can accurately decode a coded audio signal bit stream containing a mixture of frequency domain and time domain signals.

As described above, the signal coding system of the present invention is capable of efficiently coding audio signals that contain a mixture of constant-type sound and impulse-type sound signals. Also, since impulse-type sound data, which gives rise to abrupt energy changes, is coded in the time domain, so-called pre-echo noise does not occur, and the degradation of coding quality associated therewith is prevented.

In addition, since signals with little energy dispersion are selected, they can be used for a vector quantization (VQ) pre-process, utilizing the statistical bias of the spaces to generate the VQ code book.

In the signal decoding system of the present invention, the advantage is the system's capability to accurately decode a coded audio signal bit stream including a mixture of frequency domain and time domain signals.

What is claimed is:

1. A signal coding system for coding an input digital signal, comprising:

- a data accumulation means for dividing an input digital signal represented in a time domain into set time intervals, and outputting said signal;
- a discrete transform processing means for transforming a digital signal received from said data accumulation means into a digital signal represented in a frequency domain;
- a discrimination means for determining whether an input digital signal is a constant-type digital signal or an impulse-type digital signal, and for concurrently outputting an identification signal indicating the result of this determination;
- a coding means that if, based on the identification signal received from said discrimination means, said input digital signal is a constant-type signal, codes said digital signal transformed by said discrete transform processing means for representation in said frequency domain, and if said input digital signal is an impulse-type signal, codes said digital signal represented in said time domain; and
- a multiplexing means for multiplexing said digital signal coded by said coding means with said identification signal.

2. The signal coding system of claim 1, wherein said discrimination means compares the energy dispersion of said digital signal represented in the frequency domain and received from said discrete transform processing means with the energy dispersion of said digital signal represented in the time domain and received from said data accumulation means, and outputs the one of these two digital signals that has the least energy dispersion.

3. The signal coding system of claim 1, wherein said discrimination means determines the differences between the amounts of energy at various frequencies in said digital signal represented in the frequency domain and received from said discrete transform processing means, and classifies digital signals in which there are large energy differences as constant-type digital signal, and digital signals in which there are small energy differences as impulse-type digital signal.

4. A signal coding system for coding input digital signals comprising:

- a data accumulation means for dividing an input digital signal represented in a time domain into set time intervals, and outputting said signal;
- a discrimination means for determining whether a digital signal received from said data accumulation means is a constant-type digital signal or an impulse-type digital signal, and for concurrently outputting an identification signal indicating the result of this determination;
- a discrete transform processing means for transforming a digital signal received from said discrimination means into a digital signal represented in a frequency domain;
- a coding means that if, based on the identification signal received from said discrimination means, said input digital signal is a constant-type signal, codes said digital signal transformed by said discrete processing means for representation in said frequency domain, and if said input digital signal is an impulse-type signal, codes said digital signal represented in said time domain; and
- a multiplexing means for multiplexing said digital signal coded by said coding means with said identification signal.

5. The signal coding system of claim 4, wherein said discrimination means detects the difference between the immediately preceding amplitude and the present amplitude in digital signals represented in the time domain and received from said data accumulation means, and classifies digital signals in which said difference is below a predetermined value as constant-type digital signal, and those in which said difference is above said predetermined value as impulse-type digital signal.

6. The signal coding system of claim 4, wherein said discrimination means determines auto-correlation coefficients within frames of said digital signal represented in the time domain and received from said data accumulation means, and classifies digital signals having high auto-correlation as constant-type digital signal, and digital signals having low auto-correlation as impulse-type digital signal.

7. A signal decoding system for decoding a digital signal divided into set time intervals containing a mixture of digital signals represented in the frequency domain and digital signals represented in the time domain and coded in this mixed state, and also having multiplexed therein, identification signals that identify the content of each time interval as either a time domain or a frequency domain signal, comprising:

- a separation means for separating an input digital signal into said coded digital signal and said identification signal portions;

a decoding means for decoding said coded digital signal received from said separation means;

a discrimination means for determining whether a digital signal received from said decoding means is represented in the frequency domain or in the time domain, based on said identification signal received from said separation means;

an inverse discrete transform processing means for transforming a digital signal received from said discrimination means represented in the frequency domain to a digital signal represented in the time domain; and

an output means for outputting, in time series sequence, digital signals represented in the time domain received from said inverse discrete transform processing means, and digital signals represented in the time domain received from said discrimination means.

* * * * *